

SACCDMM - Curs 06

Standardul de compresie JPEG

Sl.Dr.Ing. Camelia FLOREA

JPEG

- JPEG - acronim pentru grupul de experți fotografi care a elaborat standardul „Joint Photographic Experts Group”;
 - grup de experti provenind din mai multe organizatii de standardizare
 - ISO -International Standards Organization
 - IEC -International Electrotechnical Comission
 - ITU -International Telecommunications Union

JPEG clasic

ISO/ IEC 10928-1

ITU-T Recommendation T-81

draft standard: 1991

international standard: 1992

JPEG 2000

ISO/ IEC 15444-1

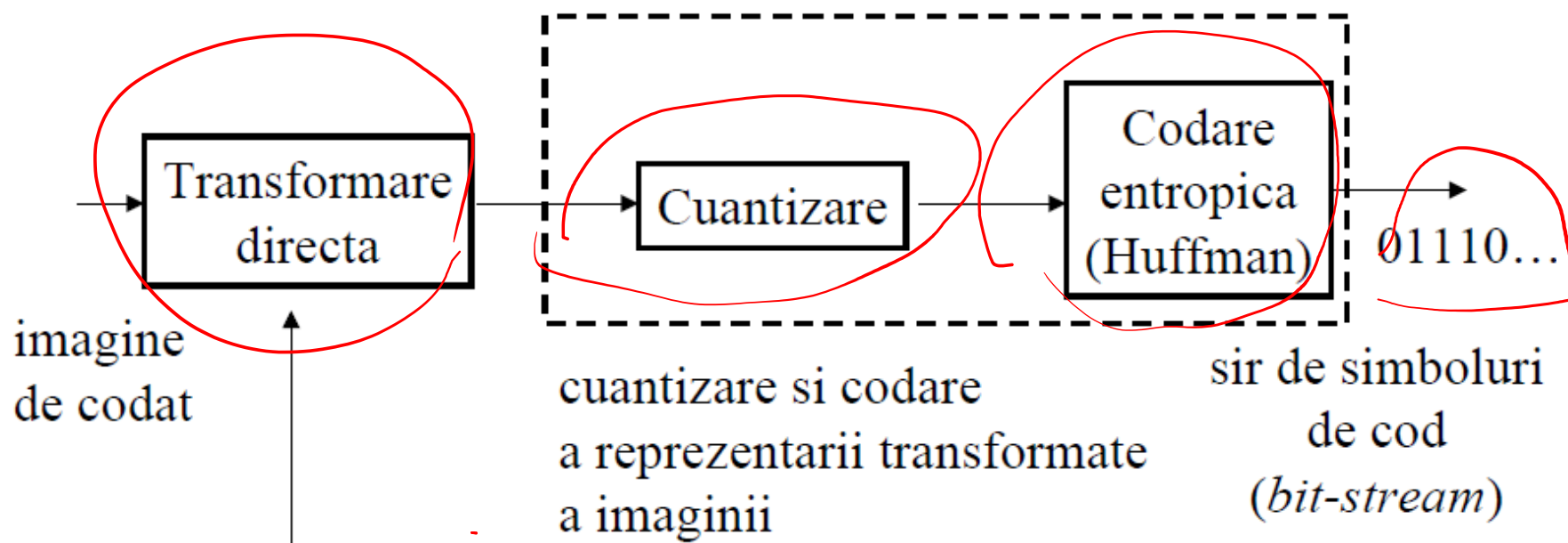
ITU-T Recommendation T-800

draft standard: 2000

international standard: Dec. 2000
(partea 1)

Schema bloc de codare

JPEG



prelucrari in suportul spatial al imaginii
prelucrari in domeniul valorilor pixelilor
sunt diferite la JPEG / JPEG2000

JPEG clasic

- JPEG este un mecanism standardizat de **compresie a imaginilor**.
 - un algoritm pentru compresia imaginilor color (cu 24-biți pe pixel) sau pe nivele de gri **ce descriu scene din lumea reală** (de exemplu, fotografii).
- astfel, JPEG operează bine pe fotografii, pe opere de artă naturaliste, sau orice scene similare;
 - însă **prezintă pierderi** în cazul imaginilor de tip **inscripții, desene sau schițe** (datorită algoritmului de compresie).

JPEG

- **Compresie „cu pierderi” bazata pe DCT** (Discrete Cosine Transform)
 - imaginea obținută prin decompresie NU este identică cu cea inițială,
 - însă, vizual diferențele sunt imperceptibile!
- JPEG este dedicat *compresiei imaginilor care vor fi vizualizate de oameni*
 - pierderile detaliilor imperceptibile vizual, sunt implementate în vederea maximizării ratei de compresie!
- ⇒ JPEG exploatează limitările ochiului uman,
 - anume faptul că schimbările mici de culoare sunt percepute cu o acuratețe mai mică decât schimbările mici de strălucire.
- Principalul câștig
 - compresii puternice, cu o rată de compresie de până la 100:1 (de obicei de 10:1 până la 20:1 fără degradare majoră).

JPEG

- Obiectivele JPEG
 - să poată fi ușor de implementat în aplicații
 - să permită alegerea gradului de compresie și/sau calitatea
 - independent de tipul imaginii
 - grad de complexitate redus
 - permite codarea secvențială
 - permite codarea progresivă
 - opțiuni de codare ierarhica

Introducere

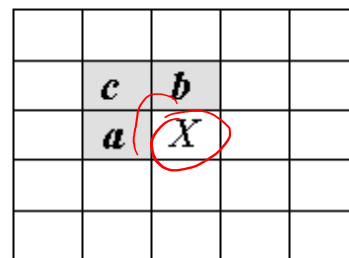
- 1988 – utilizarea DCT ca transf.
 - 1988-1990 teste, simulari, evaluari
 - 1991 – propunere de standard
 - 1992 – standard international
-
- **doua metode de compresie**
 - metoda de compresie *cu pierderi* – DCT
 - metoda de compresie *fara pierderi*

Metode de compresie JPEG

- **Metoda de compresie cu pierderi** bazată pe transformata cosinus discretă
 - este vorba de formatul JPEG „clasic”, care permite rate de compresie importante (de 10:1 până la 20:1) păstrând în același timp o foarte bună calitate a imaginii; această metodă de compresie este ireversibilă.
- **Metoda de compresie predictivă fără pierderi**
 - nu au loc pierderi de informație și este în consecință posibilă reproducerea exactă a imaginii originale, dar rata compresiei posibilă cu această metodă este mult mai modestă (aproximativ 2:1); această metodă de compresie este reversibilă.

JPEG fara pierderi

- NU folosește algoritmi de compresie prin transformări – DCT
- Factor de compresie slab
- Algoritmul **JPEG fara pierderi**:
 - codare diferentiala DPCM
 - codor Huffman sau arithmetic



- Predictorii:
 - eroarea de predicție: $e = y - X$

$$\begin{aligned}
 y &= 0 & y &= a + b - c \\
 y &= a & y &= a + \frac{b - c}{2} \\
 y &= b & y &= b + \frac{a - c}{2} \\
 y &= c & y &= \frac{a + b}{2}
 \end{aligned}$$

Modul de operare compresie pe baza de DCT

- Standardul JPEG specifică trei moduri de operare pentru compresia imaginilor:
 - **modul secvențial bazat pe DCT**
 - modul de bază de operare JPEG - cel mai popular
 - suportă doar codare cu pierderi
 - **modul progresiv bazat pe DCT**
 - se realizează un set de subimagini, fiecare subimagine fiind codată cu un set specific de coeficienți DCT.
 - codorul DCT va trebui să aibă un buffer în care datele (coeficienții DCT ai subimaginilor) să fie memorate înainte de codarea entropică.
 - **modul ierarhic**
 - pentru aplicații – când e nevoie de imagini la rezoluții multiple
 - fiecare plan este codat ca o secvență de cadre (primul cadru – versiune subesantionată a imaginii originale; cadrele următoare – cadre diferență)
 - se poate folosi: JPEG cu pierderi; JPEG fără pierderi; combinat (ultima etapă JPEG fără pierderi).

Spațiul de culoare în JPEG

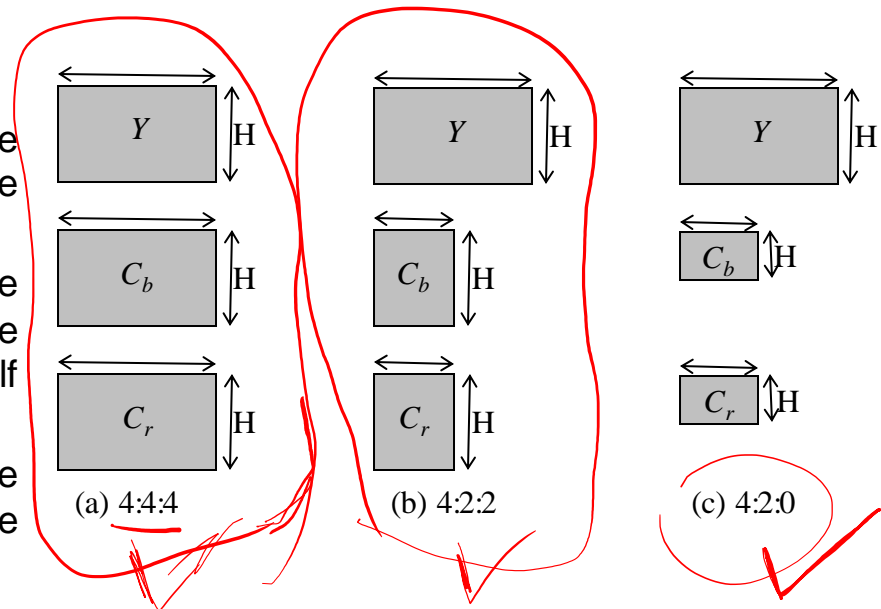
- Imaginile pot fi reprezentate pe mai multe planuri de culoare
 - cele mai multe dispozitive de scanare generează imagini pe planuri de culoare R,G,B.
 - standardul JPEG nu prezintă restricții pentru tipul imaginii de intrare
 - uzual compresia se realizează pe un format care realizează o decorelare între luminanță și cromaticitate: YUV, YCbCr, etc.
- Imagina este interpretată ca o colecție de planuri de imagine/ componente
 - maxim 255 planuri de imagine (matrice de pixeli)
 - se comprimă fiecare componentă de culoare separat
- pixeli reprezentați:
 - 8 sau 12 biti/pixel – codarea cu DCT;
 - între 2 – 16 biti/pixel – codarea fără pierderi

Subeșantionarea

- JPEG acceptă ca cele trei componente ale tripletului YUV/ YCbCr să fie **comprimate folosind rate de compresie diferite**.
- Compresia JPEG este **mai eficientă** atunci culoarea este reprezentată sub forma decorelată: **luminanță (strălucire) și crominanță**.
 - creșterea eficienței codării se explică prin faptul că ochiul este mai puțin sensibil la schimbările de culoare decât la schimbările de luminozitate
=> canalele de crominanță pot fi **codate cu pierderi mai mari** decât canalul de luminanță.
- Utilizare în mod curent: câte o valoare U și V pentru fiecare 4 valori ale componentei Y (4:2:0)
 - **se va salva 50% din spațiul utilizat** (6 valori în loc de 12) fără a afecta calitatea imaginii din punctul de vedere a percepției ochiului uman.

- There are three color format in the baseline system :

- 4:4:4** The chrominance components have identical vertical and horizontal resolution as the luminance component.
- 4:2:2** The chrominance components have the same vertical resolution as the luminance component, but the horizontal resolution is half one.
- 4:2:0** Both vertical and horizontal resolution of the chrominance component is half of the luminance component.



Dimensiunea componentelor

- **Componentele pot avea dimensiuni diferite:**
 - x_i, y_i dimensiunile pentru componenta i
 - x_j, y_j dimensiunile pentru componenta j
 - factorii de esantionare relativa (H_i, V_i) iau valori intre 1 - 4

- In format/ fișier se specifică
 - dimensiunile maxime (X, Y)
 - factorii de esantionare relativi maximi (H_{\max}, V_{\max})
 - factorii de esantionare componente (H_i, V_i)

- Dacă avem X, H_{\max} si H_i
 - putem determina x_i pentru fiecare plan
- Dacă avem Y, V_{\max} si V_i
 - putem determina y_i pentru fiecare plan

$$\frac{x_i}{x_j} = \frac{H_i}{H_j}$$

$$\frac{y_i}{y_j} = \frac{V_i}{V_j}$$

$$X = \max(x_i);$$

$$Y = \max(y_i).$$

~~$x_i = X \cdot \frac{H_i}{H_{\max}}$~~

$$x_i = \left[X \cdot \frac{H_i}{H_{\max}} \right]$$

$$y_i = \left[Y \cdot \frac{V_i}{V_{\max}} \right]$$

Intrețeserea planurilor de culoare

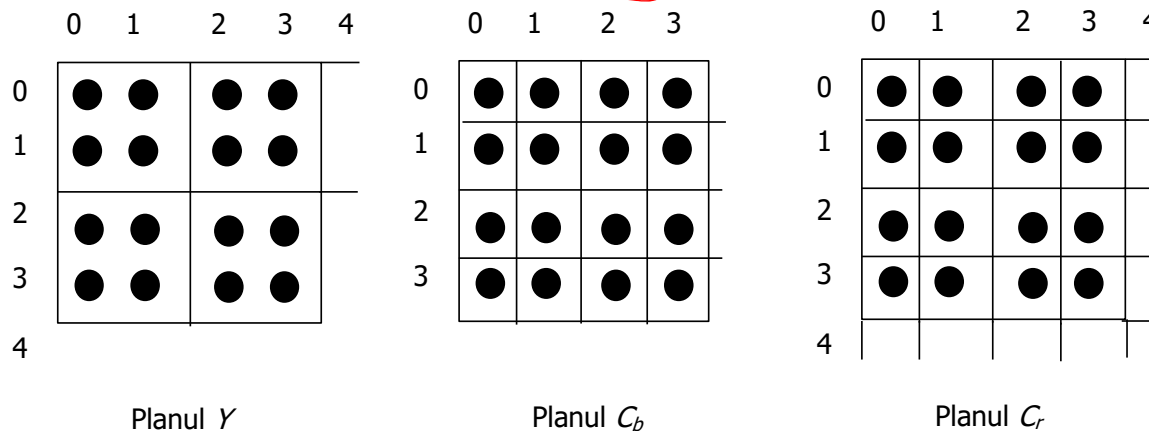
- moduri de prelucrare:
 - fara intrețesere
 - fiecare plan de culoare/componentă este prelucrată separat
 - ex. RGB - > 3 planuri de culoare de 8 biti (R,G,B)
 - cu intrețesere
 - definim un **element unitar**
 - pixel – compresia fara pierderi
 - bloc de 8x8 – compresia cu pierderi
 - **unitatea de codare minima (MCU)**
 - planul de culoare i este impartit in blocuri de dim. $H_i \times V_i$
 - selectarea subblocurilor corespondente din fiecare plan
 - cea mai mica grupare de date intrețesute

DCT

16x16
h blocuri

Exemplu de intretesere

- Spatiu de culoare YUV sau YCbCr
 - Y – 2x2 blocuri de cate 8x8 pixeli;
 - C_b , C_r – câte un bloc elementar de 8x8 pixeli
- MCU – 4 blocuri de Y; 1 bloc de C_b ; 1 bloc de C_r



$$MCU_1 = Y_{00}Y_{01}Y_{10}Y_{11}C_{b00}C_{r00}; \quad MCU_2 = Y_{02}Y_{03}Y_{12}Y_{13}C_{b01}C_{r01}$$

- numarul maxim de planuri intretesute este 4
- numarul maxim de elemente dintr-un MCU este 10
 - daca nu se satisface aceasta conditie => codare ne-intretesuta

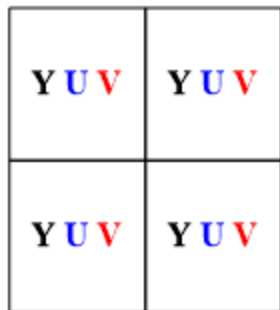
Modul secvențial bazat pe DCT

- Presupune transformarea unui bloc de imagine de dimensiune $N \times N$ din **domeniul spațial** în **domeniul coeficienților DCT** (spațiul frecvențelor) și aplicarea a catorva etape de prelucrare pentru compresie.
 - în general, valoarea lui N este egală cu 8.
- Alegerea dimensiunii de 8×8 pixeli pentru blocul de bază de este motivată de mai multe considerente:
 - **implementare hardware și software mai ușoară** datorită necesarului relativ scăzut de memorie
 - complexitatea de calcul la un bloc de 8×8 pixeli este relativ redusă
 - din punct de vedere al eficienței, alegerea unui bloc mai mare de 8×8 pixeli **nu oferă un grad de compresie semnificativ mai mare**

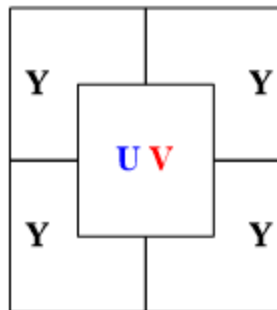
Descompunerea în blocuri

- Procedura de compresie se aplică pe rând, pe fiecare bloc de 8×8 pixeli din imagine – astfel, cele trei componente Y, U și V sunt descompuse în blocuri de această dimensiune.
- Datorită reducerii rezoluției componentelor de crominanță U și V prin subeșantionare rezultă că la 4 sub-blocuri de 2×2 pixeli ale componentei Y le corespunde câte un singur bloc 2×2 valori de ale componentelor U, respectiv V.
- Blocurile de imagine din cele trei componente sunt stocate întrețesut.
- Ordinea stocării acestora va fi:

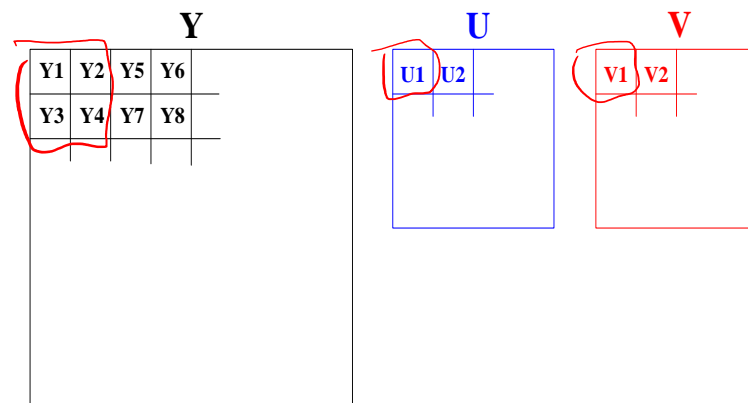
Y1, Y2, Y3, Y4, U1, V1, Y5, Y6, Y7, Y8, U2, V2 ...



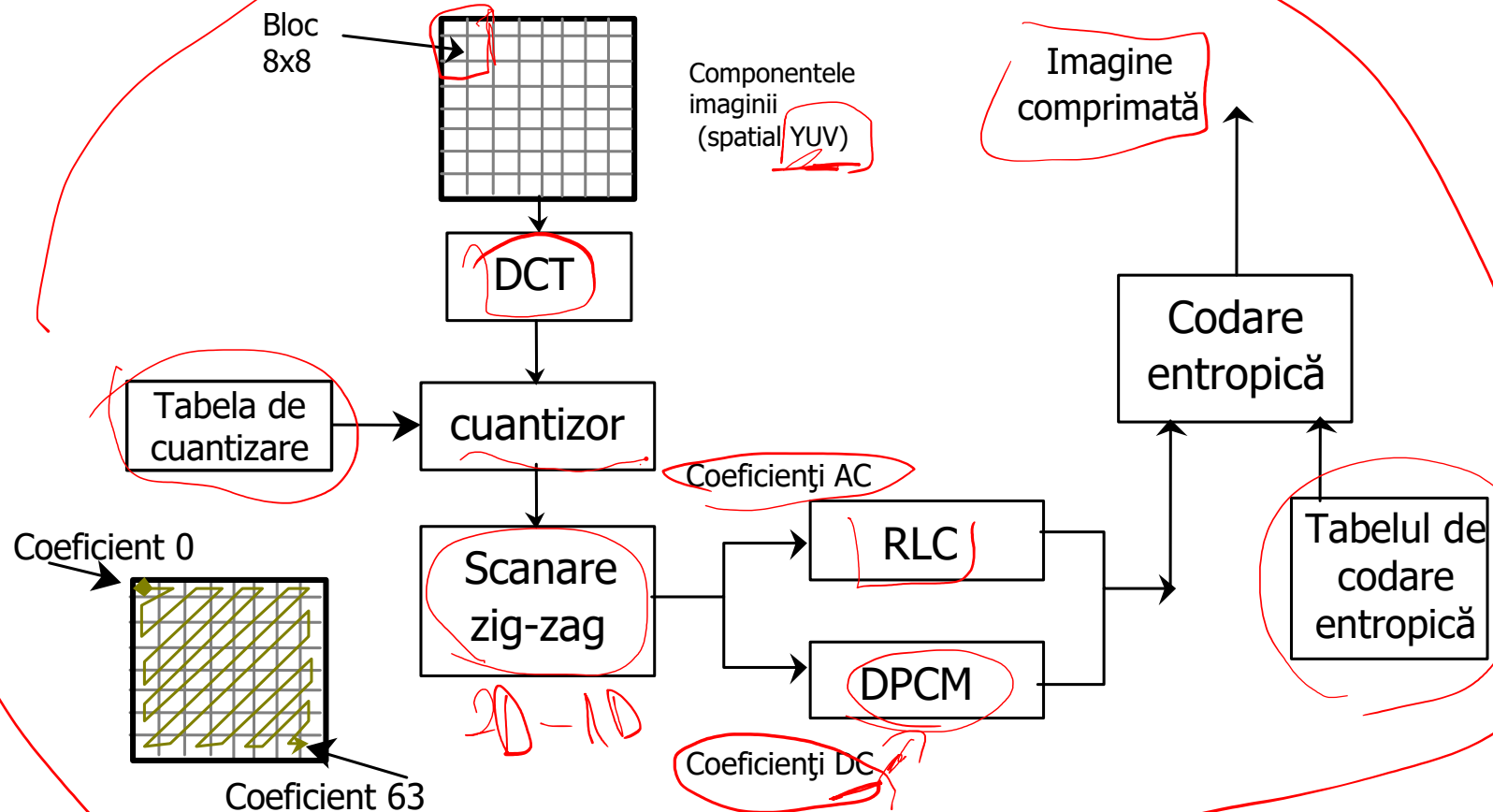
Înainte de
subeșantionare



După
subeșantionare



Codorul JPEG



Procesare bloc spre compresie

- Fiecare bloc de 8×8 pixeli este prelucrat (comprimat):
 - axarea componentelor pe zero
 - aplicarea transformatei cosinus discrete,
 - cuantizarea matricii transformate,
 - calculul coeficientului diferență DC,
 - aranjarea coeficienților DCT în zigzag,
 - codare RLE, reprezentarea șirului RLE în format binar,
 - codare entropică (codarea coeficientului DC și codarea coeficienților AC).

Axarea componentelor pe zero

- Valorile originale ale eşantioanelor semnalelor Y respectiv U şi V sunt cuprinse în domeniul $[0, 2^b-1]$, unde b reprezintă numărul de biţi/eşantion.
- Aceste valori sunt deplasate în domeniul $[-2^{b-1}, 2^{b-1}-1]$, **centrate faţă de zero** pentru a putea realiza o precizie de calcul mai mare la aplicare DCT.
- În cazul compresiei JPEG, pentru fiecare componentă în parte, vom avea $b=8$ astfel încât valorile intensităţilor din imaginea originală cuprinse în intervalul $[0, 255]$ sunt deplasate în intervalul $[-128, 127]$.

Aplicarea transformatei cosinus discrete (DCT)

- prin intermediul acestei transformări, un vector conținând date despre intensitate este transformat într-un vector care conține date despre modul de variație spațial al intensității
 - ⇒ matricea coeficienților DCT
- Elementele matricii coeficienților DCT au valorile mari concentrate în colțul din stânga-sus, iar în colțul din dreapta jos valorile sunt foarte mici - aproape nule.
 - DCT realizează o transformare a datelor în domeniul frecvență – astfel:
 - coeficienții din colțul din stânga-sus corespund frecvențelor joase - variații lente de intensitate între pixeli;
 - pe măsură ce se avansează către colțul din dreapta-jos, coeficienții corespund frecvențelor înalte - variații rapide de intensitate date de detaliile fine din imagine.
- În general într-o imagine reală frecvențele înalte au o pondere mai redusă decât cele joase, ceea ce explică valorile obținute în urma transformării.

Cuantizarea coeficienților DCT ai blocurilor

- Operația de **cuantizare** este singura în care **se pierde informație** → efecte de blocking
- Rol cuantizare – a se obține cât mai multe valori nule sau mici
→ avantajul unei compresii eficiente realizate ulterior
- Aplicare pe DCT - oferă posibilitatea de a realiza această operație astfel încât pierderile de informație să fie cât mai reduse
- Există o matrice de cuantizare pentru fiecare plan de culoare
- Trebuie realizat un echilibru rata de compresie – calitate (tehnici de proiectare psihovizuale)
 - controlul ratei de bit - mai mulți biți pentru coeficienții cu varianță mare
- Dacă valorile din tabelele standard de cuantizare sunt divizate cu 2, imaginea reconstruită este de obicei identică vizual cu imaginea sursă

| | | | | | | | |
|----|----|----|----|-----|-----|-----|-----|
| 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 |

Tabelele de cuantizare **Y** luminanță

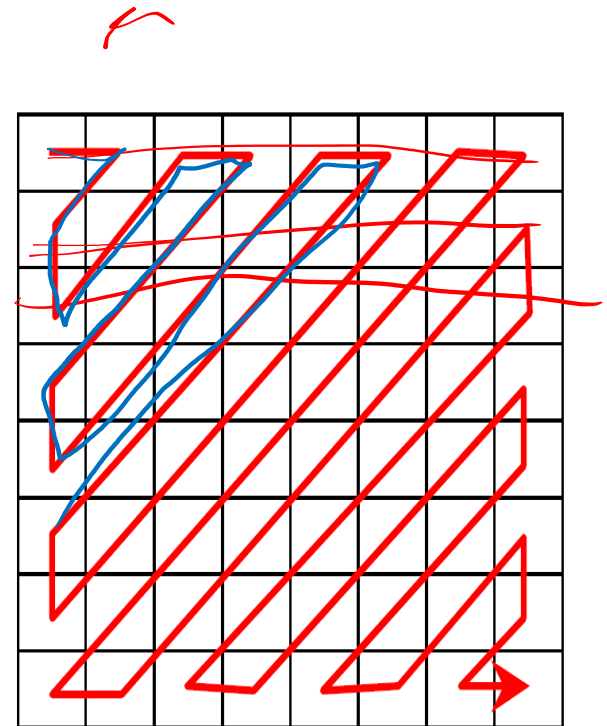
| | | | | | | | |
|----|----|----|----|----|----|----|----|
| 17 | 18 | 24 | 47 | 99 | 99 | 99 | 99 |
| 18 | 21 | 26 | 66 | 99 | 99 | 99 | 99 |
| 24 | 26 | 56 | 99 | 99 | 99 | 99 | 99 |
| 47 | 66 | 99 | 99 | 99 | 99 | 99 | 99 |
| 99 | 99 | 99 | 99 | 99 | 99 | 99 | 99 |
| 99 | 99 | 99 | 99 | 99 | 99 | 99 | 99 |
| 99 | 99 | 99 | 99 | 99 | 99 | 99 | 99 |
| 99 | 99 | 99 | 99 | 99 | 99 | 99 | 99 |

Tabelele de cuantizare **cr** cromatică

$$U_{\text{dct}_Q}(i,j) = [U_{\text{dct}}(i,j) / Q(i,j)], \text{ unde } i, j = \{0 \dots 7\}$$

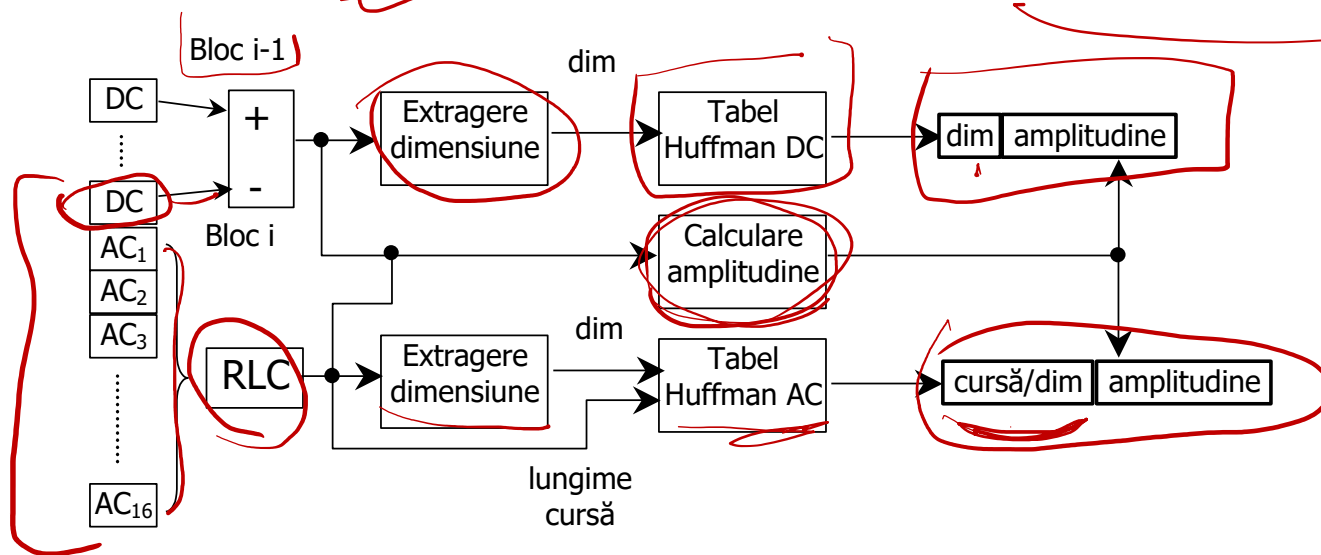
Aranjarea blocului în zigzag

- Ordinea în care este parcursă matricea în vederea codării coeficienților de tip AC se alege în așa fel încât să se profite cât mai bine de distribuția valorilor coeficienților.
- Se urmărește gruparea valorilor nule în șiruri cât mai lungi deoarece acest fapt permite o codare cât mai eficientă (compresie maximă).
- Deoarece valorile nenule sunt concentrate într-un colț al matricii, o parcurgere normală de tipul linie cu linie nu este eficientă, de aceea se preferă o parcurgere în zigzag.



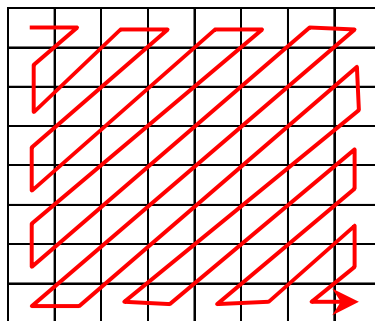
Codarea entropică

- variante Huffman, aritmetică +
- DPCM (DC), RLC (AC)
- Huffman - se pot folosi specificațiile standardului
- limitări:
 - cuvinte de cod < 16 biți
 - cuvinte de cod FF trebuie urmate de un cuvânt de cod 00



Codarea Huffman

- coeficientii DC – (dim, amplitudine) 16 categorii
- Coeficientii AC [-1023, 1023]
 - 10 categorii de amplitudine
 - scanarea zig-zag
 - cuvântul de cod – (lungime_cursa/dim, amp)
 - Lungime cursa > 15 codul (15/0) – partajeaza de cate ori este nevoie
 - Daca dupa un AC nenul toti sunt nuli – EOB (0/0)



Calcul coeficient diferență DC (DPCM – Differential Pulse Code Modulation)

- Coeficientul diferență DC este obținut prin scăderea coeficientului DC al blocului anterior procesat din coeficientul DC curent.
- Această operație este echivalentă cu o codare diferențială a coeficienților DC și se face pentru a se obține coeficienți de codat cât mai mici posibil, fiind aplicabilă tuturor coeficienților DC cu excepția primului coeficient DC din imagine.
- Codul unei valori se obține prin concatenarea unui cod pentru clasa din care face parte și un cod corespunzător valorii acesteia.
- Exemplu - cod de clasă 5 (SSSS=0101) citind în tabelă se găsește codul de 3 biți 110.
 - după codul de clasă trebuie adăugată valoarea coeficientului diferența DC - cei mai puțin semnificativi SSSS biți din valoarea în cazul valorilor pozitive sau din valoarea în complement față de 2 minus 1 dacă aceasta este negativă.

| Clasa (SSSS) - 4 biți | Valoare diferență |
|-----------------------|--------------------------------|
| 0 (0000) | 0 |
| 1 (0001) | -1, 1 |
| 2 (0010) | -3, -2, 2, 3 |
| 3 (0011) | -7,...,-4, 4,...,7 |
| 4 (0100) | -15,...,-8, 8,...,15 |
| 5 (0101) | -31,...,-16, 16,...,31 |
| 6 (0110) | -63,...,-32, 32,...,63 |
| 7 (0111) | -127,...,-64, 64,...,127 |
| 8 (1000) | -255,...,-128, 128,...,255 |
| 9 (1001) | -511,...,-256, 256,...,511 |
| 10 (1010) | -1023,...,-512, 512,...,1023 |
| 11 (1011) | -2047,...,-1024, 1024,...,2047 |

Codare coeficient diferenta DC

- Codare valoare 21 ce aparține componentei Y
 - clasa 5 - pentru care codul este 110.
 - valoare pozitivă – se adauga cei mai puțin semnificativi 5 biți ai acesteia: 10101
 - ⇒ reprezentare binară 11010101 - cod complet (cod clasa + cod valoare)
- Codare valoare -21 ce aparține componentei Y
 - aceeași clasa, clasa 5 codul este 110
 - valoare negativă - se reprezintă în complement față de 2 minus 1 (21 complement față de 2 va fi 1111111110101 se scade 1 și se păstrează cei mai puțin semnificativi 5 biți) - adică 01010.
 - Codul complet în acest caz: 11001010.

0...000010101

Tabel 1. Tabel pt diferența coeficienților DC a luminanței

| Category | Code length | Code word |
|----------|-------------|------------|
| 0 | 2 | 00 |
| 1 | 3 | 010 |
| 2 | 3 | 011 |
| 3 | 3 | 100 |
| 4 | 3 | 101 |
| <u>5</u> | <u>3</u> | <u>110</u> |
| 6 | 4 | 1110 |
| 7 | 5 | 11110 |
| 8 | 6 | 111110 |
| 9 | 7 | 1111110 |
| 10 | 8 | 11111110 |
| 11 | 9 | 111111110 |

Tabel 2. Tabel pt diferența coeficienților DC a crominanței

| Category | Code length | Code word |
|----------|-------------|-------------|
| 0 | 2 | 00 |
| 1 | 2 | 01 |
| 2 | 2 | 10 |
| 3 | 3 | 110 |
| 4 | 4 | 1110 |
| 5 | 5 | 11110 |
| 6 | 6 | 111110 |
| 7 | 7 | 1111110 |
| 8 | 8 | 11111110 |
| 9 | 9 | 111111110 |
| 10 | 10 | 1111111110 |
| 11 | 11 | 11111111110 |

Codare RLE (Run Length Encoded)

- După scanare în zigzag coeficienții DCT cuantizați - unui vector cu multe zerouri
 - coeficientul diferența DC nu se codează RLC
 - coeficienții AC cuantizați se codează RLE considerând numărul de zerouri consecutive
- **Vectorul RLE** conține perechi - **numărul de zerouri** care preced valoarea coeficientului AC nenul, urmată de valoarea **coeficientului AC nenul**
- Fiecare coeficient - număr întreg în domeniul -1023 și +1023 (valori obținute analizând cazul cel mai defavorabil în modul de operare DCT).
 - în cazul codării "normale" reprezentare pe 11 biți (primul bit pentru semn, următorii 10 biți - valoarea)
 - duce practic la o reprezentare pe mai mulți biți decât în cazul imaginii necomprimată (se folosesc 11 biți pentru un coeficient în loc de 8 biți pentru un pixel).
- Coeficienții din vectorul **RLE** sunt reprezentați prin **șiruri binare de lungime variabilă**
 - reprezentare sub forma: **lungime cursă/identificator clasă, valoare coeficient**
 - Lungime cursă - numărul de zerouri care preced valoarea coeficientului AC nenul
 - Identificator clasă ne arată lungimea în biți a șirului binar care dă valoarea coeficientului
 - Valoare coeficientului nenul
 - Valorile pozitive - încep cu un 1 codate în maniera binară obișnuită.
 - Numerele negative - încep cu un 0, fiind reprezentate în complement față de 2 minus 1.

Codare coeficienti AC

(lung cursă, ValCoefAc)

- **Identificator clasa** (SSSS) - clasa valorii coeficientului diferit de zero obținută similar ca în cazul coeficienților DC
- **Lungime cursă** (RRRR) - numărul de zerouri - valoare între 0 și 15 care poate fi reprezentată pe 4 biți
- Tabele Huffman continand coduri pentru **lungime cursă/identificator clasa**
 - pentru componenta de luminanta (Tabel 3)
 - pentru componentele de croma (Tabel 4)
- Poziția codului în tabela
 - $INDEX = 11 \times RRRR + SSSS$.
- Se rezerva codurile
 - cu RRRR=0 și SSSS=0 pentru simbolul special EOB
 - cu RRRR=15 și SSSS=0 pentru simbolul special ZRL.
- **Exemplu**, codificare (1,-4) din componentei Y
 - SSSS = 3 și RRRR = 1, valoarea aparține componentei Y - se va utiliza Tabel 3 din anexe => deci codul clasei este 1111001 reprezentat pe 7 biți.
 - Valoare coeficient nenul este -4 (1111111111111100 în complement față de 2), valoarea minus 1 este 1111111111111011, deci se vor adăuga cei mai puțin semnificativi 3 biți adică 011.
 - Codul complet al simbolului (1,-4) este 1111001011.

| Clasa (SSSS) - 4 biți | Valoare |
|-----------------------|------------------------------|
| 1 (0001) | -1, 1 |
| 2 (0010) | -3, -2, 2, 3 |
| 3 (0011) | -7,...,-4, 4,...,7 |
| 4 (0100) | -15,...,-8, 8,...,15 |
| 5 (0101) | -31,...,-16, 16,...,31 |
| 6 (0110) | -63,...,-32, 32,...,63 |
| 7 (0111) | -127,...,-64, 64,...,127 |
| 8 (1000) | -255,...,-128, 128,...,255 |
| 9 (1001) | -511,...,-256, 256,...,511 |
| 10 (1010) | -1023,...,-512, 512,...,1023 |

Tabel 3. Tabel pt coeficienții AC a luminanței (part 1/4)

| R / S (Run/Size) | Lungime cod | Cuvant cod |
|------------------|-------------|-----------------|
| 0/0 (EOB) | 4 | 1010 |
| 0/1 | 2 | 00 |
| 0/2 | 2 | 01 |
| 0/3 | 3 | 100 |
| 0/4 | 4 | 1011 |
| 0/5 | 5 | 11010 |
| 0/6 | 7 | 1111000 |
| 0/7 | 8 | 1111000 |
| 0/8 | 10 | 111110110 |
| 0/9 | 16 | 111111110000010 |
| 0/A | 16 | 111111110000011 |
| 1/1 | 4 | 1100 |
| 1/2 | 5 | 11011 |
| 1/3 | 7 | 1111001 |
| 1/4 | 9 | 11110110 |
| 1/5 | 11 | 1111110110 |
| 1/6 | 16 | 111111110000100 |
| 1/7 | 16 | 111111110000101 |
| 1/8 | 16 | 111111110000110 |
| 1/9 | 16 | 111111110000111 |
| 1/A | 16 | 111111110001000 |
| 2/1 | 5 | 11100 |
| 2/2 | 8 | 1111001 |
| 2/3 | 10 | 111110111 |
| 2/4 | 12 | 11111110100 |
| 2/5 | 16 | 111111110001001 |
| 2/6 | 16 | 111111110001010 |
| 2/7 | 16 | 111111110001011 |
| 2/8 | 16 | 111111110001100 |
| 2/9 | 16 | 111111110001101 |
| 2/A | 16 | 111111110001110 |
| 3/1 | 6 | 111010 |
| 3/2 | 9 | 11110111 |
| 3/3 | 12 | 11111110101 |
| 3/4 | 16 | 111111110001111 |
| 3/5 | 16 | 111111110010000 |
| 3/6 | 16 | 111111110010001 |
| 3/7 | 16 | 111111110010010 |
| 3/8 | 16 | 111111110010011 |
| 3/9 | 16 | 111111110010100 |
| 3/A | 16 | 111111110010101 |

Tabel 3. (part 2/4)

| R / S (Run/Size) | Lungime cod | Cuvant cod |
|------------------|-------------|-----------------|
| 4/1 | 6 | 111011 |
| 4/2 | 10 | 1111111000 |
| 4/3 | 16 | 111111110001010 |
| 4/4 | 16 | 111111110001011 |
| 4/5 | 16 | 111111110001000 |
| 4/6 | 16 | 111111110001001 |
| 4/7 | 16 | 111111110001010 |
| 4/8 | 16 | 111111110001011 |
| 4/9 | 16 | 111111110001100 |
| 4/A | 16 | 111111110001101 |
| 5/1 | 7 | 1111010 |
| 5/2 | 11 | 11111110111 |
| 5/3 | 16 | 111111110001110 |
| 5/4 | 16 | 111111110001111 |
| 5/5 | 16 | 111111110100000 |
| 5/6 | 16 | 111111110100001 |
| 5/7 | 16 | 111111110100010 |
| 5/8 | 16 | 111111110100011 |
| 5/9 | 16 | 111111110100100 |
| 5/A | 16 | 111111110100101 |
| 6/1 | 7 | 1111011 |
| 6/2 | 12 | 11111110110 |
| 6/3 | 16 | 111111110100110 |
| 6/4 | 16 | 111111110100111 |
| 6/5 | 16 | 111111110101000 |
| 6/6 | 16 | 111111110101001 |
| 6/7 | 16 | 111111110101010 |
| 6/8 | 16 | 111111110101011 |
| 6/9 | 16 | 111111110101100 |
| 6/A | 16 | 111111110101101 |
| 7/1 | 8 | 1111010 |
| 7/2 | 12 | 11111110111 |
| 7/3 | 16 | 111111110101110 |
| 7/4 | 16 | 111111110101111 |
| 7/5 | 16 | 111111110110000 |
| 7/6 | 16 | 111111110110001 |
| 7/7 | 16 | 111111110110010 |
| 7/8 | 16 | 111111110110011 |
| 7/9 | 16 | 111111110110100 |
| 7/A | 16 | 111111110110101 |
| 8/1 | 9 | 11111000 |
| 8/2 | 15 | 11111111000000 |

Tabel 3. (part 3/4)

| R / S (Run/Size) | Lungime cod | Cuvant cod |
|------------------|-------------|-----------------|
| 8/3 | 16 | 111111110110110 |
| 8/4 | 16 | 111111110110111 |
| 8/5 | 16 | 111111110111000 |
| 8/6 | 16 | 111111110111001 |
| 8/7 | 16 | 111111110111010 |
| 8/8 | 16 | 111111110111011 |
| 8/9 | 16 | 111111110111100 |
| 8/A | 16 | 111111110111101 |
| 9/1 | 9 | 111111001 |
| 9/2 | 16 | 111111110111110 |
| 9/3 | 16 | 111111110111111 |
| 9/4 | 16 | 111111110000000 |
| 9/5 | 16 | 111111110000001 |
| 9/6 | 16 | 111111110000010 |
| 9/7 | 16 | 111111110000011 |
| 9/8 | 16 | 111111110000100 |
| 9/9 | 16 | 111111110000101 |
| 9/A | 16 | 111111110000110 |
| A/1 | 9 | 111111010 |
| A/2 | 16 | 111111110000111 |
| A/3 | 16 | 111111110001000 |
| A/4 | 16 | 111111110001001 |
| A/5 | 16 | 111111110001010 |
| A/6 | 16 | 111111110001011 |
| A/7 | 16 | 111111110001100 |
| A/8 | 16 | 111111110001101 |
| A/9 | 16 | 111111110001110 |
| A/A | 16 | 111111110001111 |
| B/1 | 10 | 1111111001 |
| B/2 | 16 | 11111111010000 |
| B/3 | 16 | 111111110100001 |
| B/4 | 16 | 111111110100010 |
| B/5 | 16 | 111111110100011 |
| B/6 | 16 | 111111110100100 |
| B/7 | 16 | 111111110100101 |
| B/8 | 16 | 111111110100110 |
| B/9 | 16 | 111111110100111 |
| B/A | 16 | 111111110100100 |
| C/1 | 10 | 111111010 |
| C/2 | 16 | 11111111011001 |
| C/3 | 16 | 11111111011010 |
| C/4 | 16 | 11111111011011 |

Tabel 3. (part 4/4)

| R / S (Run/Size) | Lungime cod | Cuvant cod |
|------------------|-------------|-----------------|
| C/5 | 16 | 11111111011100 |
| C/6 | 16 | 11111111011101 |
| C/7 | 16 | 11111111011110 |
| C/8 | 16 | 11111111011111 |
| C/9 | 16 | 11111111100000 |
| C/A | 16 | 11111111100001 |
| D/1 | 11 | 1111111000 |
| D/2 | 16 | 11111111100010 |
| D/3 | 16 | 11111111100011 |
| D/4 | 16 | 111111111000100 |
| D/5 | 16 | 111111111000101 |
| D/6 | 16 | 111111111000110 |
| D/7 | 16 | 111111111000111 |
| D/8 | 16 | 11111111101000 |
| D/9 | 16 | 11111111101001 |
| D/A | 16 | 11111111101010 |
| E/1 | 16 | 11111111101011 |
| E/2 | 16 | 111111111010100 |
| E/3 | 16 | 111111111010101 |
| E/4 | 16 | 111111111010110 |
| E/5 | 16 | 111111111010111 |
| E/6 | 16 | 11111111100000 |
| E/7 | 16 | 11111111100001 |
| E/8 | 16 | 11111111100010 |
| E/9 | 16 | 11111111100011 |
| E/A | 16 | 11111111101000 |
| F/0 (ZRL) | 11 | 1111111001 |
| F/1 | 16 | 11111111101010 |
| F/2 | 16 | 111111111010101 |
| F/3 | 16 | 11111111101011 |
| F/4 | 16 | 11111111101000 |
| F/5 | 16 | 11111111101001 |
| F/6 | 16 | 11111111101010 |
| F/7 | 16 | 11111111101011 |
| F/8 | 16 | 11111111101100 |
| F/9 | 16 | 11111111101101 |
| F/A | 16 | 11111111101110 |

Tabel 4. Tabel pt coeficientii AC a crominantei (part 1/4)

| R / S (Run/Size) | Lungime cod | Cuvant cod |
|------------------|-------------|------------------|
| 0/0 (EOB) | 2 | 00 |
| 0/1 | 2 | 01 |
| 0/2 | 3 | 100 |
| 0/3 | 4 | 1010 |
| 0/4 | 5 | 11000 |
| 0/5 | 5 | 11001 |
| 0/6 | 6 | 111000 |
| 0/7 | 7 | 1111000 |
| 0/8 | 9 | 11110100 |
| 0/9 | 10 | 111110110 |
| 0/A | 12 | 11111110100 |
| 1/1 | 4 | 1011 |
| 1/2 | 6 | 111001 |
| 1/3 | 8 | 11110110 |
| 1/4 | 9 | 111110101 |
| 1/5 | 11 | 11111110110 |
| 1/6 | 12 | 111111110101 |
| 1/7 | 16 | 111111110001000 |
| 1/8 | 16 | 111111110001001 |
| 1/9 | 16 | 111111110001010 |
| 1/A | 16 | 111111110001011 |
| 2/1 | 5 | 11010 |
| 2/2 | 8 | 11110111 |
| 2/3 | 10 | 1111110111 |
| 2/4 | 12 | 111111110110 |
| 2/5 | 15 | 111111111000010 |
| 2/6 | 16 | 1111111110001100 |
| 2/7 | 16 | 1111111110001101 |
| 2/8 | 16 | 1111111110001110 |
| 2/9 | 16 | 1111111110001111 |
| 2/A | 16 | 1111111110010000 |
| 3/1 | 5 | 11011 |
| 3/2 | 8 | 11111000 |
| 3/3 | 10 | 111111000 |
| 3/4 | 12 | 111111110111 |
| 3/5 | 16 | 1111111110010001 |
| 3/6 | 16 | 1111111110010010 |
| 3/7 | 16 | 1111111110010011 |
| 3/8 | 16 | 1111111110010100 |
| 3/9 | 16 | 1111111110010101 |
| 3/A | 16 | 1111111110010110 |
| 4/1 | 6 | 111010 |

Tabel 4. (part 2/4)

| R / S (Run/Size) | Lungime cod | Cuvant cod |
|------------------|-------------|------------------|
| 4/2 | 9 | 111110110 |
| 4/3 | 16 | 1111111110010111 |
| 4/4 | 16 | 1111111110011000 |
| 4/5 | 16 | 1111111110011001 |
| 4/6 | 16 | 1111111110011010 |
| 4/7 | 16 | 1111111110011011 |
| 4/8 | 16 | 1111111110011100 |
| 4/9 | 16 | 1111111110011101 |
| 4/A | 16 | 1111111110011110 |
| 5/1 | 6 | 111011 |
| 5/2 | 10 | 1111111001 |
| 5/3 | 16 | 1111111110011111 |
| 5/4 | 16 | 1111111110100000 |
| 5/5 | 16 | 1111111110100001 |
| 5/6 | 16 | 1111111110100010 |
| 5/7 | 16 | 1111111110100011 |
| 5/8 | 16 | 1111111110100100 |
| 5/9 | 16 | 1111111110100101 |
| 5/A | 16 | 1111111110100110 |
| 6/1 | 7 | 1111001 |
| 6/2 | 11 | 111111110111 |
| 6/3 | 16 | 1111111110100111 |
| 6/4 | 16 | 1111111110101000 |
| 6/5 | 16 | 1111111110101001 |
| 6/6 | 16 | 1111111110101010 |
| 6/7 | 16 | 1111111110101011 |
| 6/8 | 16 | 1111111110101100 |
| 6/9 | 16 | 1111111110101101 |
| 6/A | 16 | 1111111110101110 |
| 7/1 | 7 | 1111010 |
| 7/2 | 11 | 11111111000 |
| 7/3 | 16 | 1111111110101111 |
| 7/4 | 16 | 1111111110110000 |
| 7/5 | 16 | 1111111110110001 |
| 7/6 | 16 | 1111111110110010 |
| 7/7 | 16 | 1111111110110011 |
| 7/8 | 16 | 1111111110110100 |
| 7/9 | 16 | 1111111110110101 |
| 7/A | 16 | 1111111110110110 |
| 8/1 | 8 | 11111001 |
| 8/2 | 16 | 1111111110110111 |
| 8/3 | 16 | 1111111110111000 |

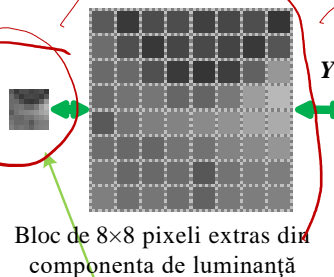
Tabel 4. (part 3/4)

| R / S (Run/Size) | Lungime cod | Cuvant cod |
|------------------|-------------|------------------|
| 8/4 | 16 | 1111111110111001 |
| 8/5 | 16 | 1111111110111010 |
| 8/6 | 16 | 1111111110111011 |
| 8/7 | 16 | 1111111110111100 |
| 8/8 | 16 | 1111111110111101 |
| 8/9 | 16 | 1111111110111110 |
| 8/A | 16 | 1111111110111111 |
| 9/1 | 9 | 111110111 |
| 9/2 | 16 | 1111111110000000 |
| 9/3 | 16 | 1111111110000001 |
| 9/4 | 16 | 1111111110000010 |
| 9/5 | 16 | 1111111110000011 |
| 9/6 | 16 | 1111111110000100 |
| 9/7 | 16 | 1111111110000101 |
| 9/8 | 16 | 1111111110000110 |
| 9/9 | 16 | 1111111110000111 |
| 9/A | 16 | 1111111110010000 |
| A/1 | 9 | 111111000 |
| A/2 | 16 | 1111111110010001 |
| A/3 | 16 | 1111111110010010 |
| A/4 | 16 | 1111111110010011 |
| A/5 | 16 | 1111111110010100 |
| A/6 | 16 | 1111111110010101 |
| A/7 | 16 | 1111111110010110 |
| A/8 | 16 | 1111111110010111 |
| A/9 | 16 | 1111111110100000 |
| A/A | 16 | 1111111110100001 |
| B/1 | 9 | 11111001 |
| B/2 | 16 | 1111111110100010 |
| B/3 | 16 | 1111111110100011 |
| B/4 | 16 | 1111111110100100 |
| B/5 | 16 | 1111111110100101 |
| B/6 | 16 | 1111111110100110 |
| B/7 | 16 | 1111111110100111 |
| B/8 | 16 | 1111111110110000 |
| B/9 | 16 | 1111111110110001 |
| B/A | 16 | 1111111110110010 |
| C/1 | 9 | 11111010 |
| C/2 | 16 | 1111111110110101 |
| C/3 | 16 | 1111111110110100 |
| C/4 | 16 | 1111111110110101 |
| C/5 | 16 | 1111111110110110 |

Tabel 4. (part 4/4)

| R / S (Run/Size) | Lungime cod | Cuvant cod |
|------------------|-------------|------------------|
| C/6 | 16 | 1111111110111111 |
| C/7 | 16 | 1111111111000000 |
| C/8 | 16 | 1111111111000001 |
| C/9 | 16 | 1111111111000010 |
| C/A | 16 | 1111111111000011 |
| D/1 | 11 | 1111111001 |
| D/2 | 16 | 1111111111001000 |
| D/3 | 16 | 1111111111001001 |
| D/4 | 16 | 1111111111001010 |
| D/5 | 16 | 1111111111001011 |
| D/6 | 16 | 1111111111010000 |
| D/7 | 16 | 1111111111010001 |
| D/8 | 16 | 1111111111010010 |
| D/9 | 16 | 1111111111010011 |
| D/A | 16 | 1111111111010100 |
| E/1 | 14 | 11111111100000 |
| E/2 | 16 | 1111111111010101 |
| E/3 | 16 | 1111111111010110 |
| E/4 | 16 | 1111111111010111 |
| E/5 | 16 | 1111111111100000 |
| E/6 | 16 | 1111111111100001 |
| E/7 | 16 | 1111111111100010 |
| E/8 | 16 | 1111111111100011 |
| E/9 | 16 | 1111111111100100 |
| E/A | 16 | 1111111111100101 |
| F/0 (ZRL) | 10 | 1111111010 |
| F/1 | 15 | 111111111000011 |
| F/2 | 16 | 1111111111010100 |
| F/3 | 16 | 1111111111010101 |
| F/4 | 16 | 1111111111100000 |
| F/5 | 16 | 1111111111100001 |
| F/6 | 16 | 1111111111100010 |
| F/7 | 16 | 1111111111100011 |
| F/8 | 16 | 1111111111100100 |
| F/9 | 16 | 1111111111100101 |
| F/A | 16 | 1111111111100110 |

- Extragerea unui bloc de 8x8 pixeli din componenta Y și aplicarea pașilor pentru compresie



| | | | | | | | |
|----|----|----|----|----|----|----|----|
| 4 | 3 | 2 | 4 | 6 | 10 | 13 | 16 |
| 3 | 3 | 3 | 5 | 7 | 14 | 15 | 14 |
| 2 | 3 | 4 | 6 | 10 | 14 | 17 | 14 |
| 4 | 5 | 6 | 9 | 13 | 22 | 20 | 16 |
| 6 | 7 | 10 | 13 | 17 | 28 | 26 | 19 |
| 10 | 14 | 14 | 22 | 28 | 26 | 28 | 23 |
| 13 | 15 | 17 | 20 | 26 | 28 | 30 | 26 |
| 16 | 14 | 14 | 16 | 19 | 23 | 26 | 25 |

Tabela de cuantizare

| | | | | | | | |
|----|----|----|----|----|----|----|----|
| 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 |
| 08 | 09 | 10 | 11 | 12 | 13 | 14 | 15 |
| 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 |
| 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 |
| 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 |
| 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 |
| 56 | 57 | 58 | 59 | 60 | 61 | 62 | 63 |

Scanare în Zig-Zag

| | | | | | | | |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 90 | 57 | 76 | 77 | 64 | 71 | 78 | 53 |
| 109 | 79 | 56 | 70 | 73 | 68 | 57 | 85 |
| 106 | 105 | 80 | 58 | 53 | 62 | 97 | 151 |
| 107 | 115 | 118 | 102 | 100 | 120 | 157 | 184 |
| 89 | 115 | 127 | 131 | 138 | 146 | 162 | 171 |
| 109 | 104 | 102 | 116 | 114 | 131 | 137 | 149 |
| 119 | 103 | 105 | 108 | 87 | 115 | 124 | 125 |
| 127 | 111 | 118 | 110 | 98 | 120 | 125 | 129 |

Y_s

| | | | | | | | |
|-----|-----|-----|-----|-----|-----|-----|-----|
| -38 | -71 | -52 | -51 | -64 | -57 | -50 | -75 |
| -19 | -49 | -72 | -58 | -55 | -60 | -71 | -43 |
| -22 | -23 | -48 | -70 | -75 | -66 | -31 | 23 |
| -21 | -13 | -10 | -26 | -28 | -8 | 29 | 56 |
| -39 | -13 | -1 | 3 | 10 | 18 | 34 | 43 |
| -19 | -24 | -26 | -12 | -14 | 3 | 9 | 21 |
| -9 | -25 | -23 | -20 | -41 | -13 | -4 | -3 |
| -1 | -17 | -10 | -18 | -30 | -8 | -3 | 1 |

Y_{DCr}

| | | | | | | | |
|---------|--------|--------|--------|--------|-------|-------|-------|
| -181.12 | -58.32 | 73.49 | -18.54 | 7.62 | 11.75 | 9.20 | -7.35 |
| -134.15 | 30.04 | 12.30 | 3.86 | 15.69 | -5.92 | -2.77 | 6.69 |
| -107.32 | 70.10 | -19.33 | 33.69 | 6.99 | 23.20 | 10.06 | -1.74 |
| 6.09 | -11.82 | -55.59 | 21.18 | -3.29 | 15.1 | -2.20 | -0.31 |
| 53.62 | -23.28 | -27.78 | -7.03 | -19.12 | 11.52 | 0.54 | 10.92 |
| -6.22 | -2.55 | 20.13 | -10.91 | -12.62 | 15.78 | 1.67 | 2.66 |
| -7.90 | -3.07 | 15.56 | -9.11 | -3.03 | -0.86 | -0.41 | 3.14 |
| -8.01 | 0.18 | -0.99 | -11.05 | -7.15 | -0.77 | -2.87 | 1.97 |

Y_Q

| | | | | | | | |
|-----|----|----|----|----|---|---|---|
| -45 | 19 | 37 | 5 | 1 | 1 | 1 | 0 |
| -45 | 10 | 4 | 1 | 2 | 0 | 0 | 0 |
| -54 | 23 | -5 | 6 | 1 | 2 | 1 | 0 |
| 2 | -2 | -9 | 2 | 0 | 1 | 0 | 0 |
| 9 | -3 | -3 | -1 | -1 | 0 | 0 | 1 |
| -1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| -1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| -1 | 0 | 0 | -1 | 0 | 0 | 0 | 0 |

Y_{ZZ}

[-45 -19 -45 -54 10 37 -5 4 23 29 -2 -5 1 1 1 2 6 -9 -3 -1 -10 -3 2 1 0 1 0 0 2 0 -1 10 -10 10 -1 1 1 0 0 0 0 0 0 -1 0 1 0 0 1 0 0 0 0 0 0 0 0 0]

Y_{RLC}

[-45 0 -19 0 -45 0 -54 0 10 0 37 0 -5 0 4 0 23 0 29 0 -2 0 -5 0 1 0 1 2 6 -9 -3 -1 -10 -3 2 1 0 1 0 2 0 6 0 -9 0 -3 0 -10 -1 -3 0 2 0 1 1 2 2 1 -10 11 -1 1 1 -1 0 0 1 2 1 1 2 1 0 0]

Decodorul JPEG

- 1) Scanarea fișierului JPEG, decodarea antetului/ delimitatori/ flux de date.
 - 2) Decodarea Huffman a coeficienților (se folosesc tabelele din segmentul DHT din antet);
 - 3) Reordonarea zigzag a elementelor din fiecare bloc de 8×8;
 - 4) Decuantizarea elementelor din fiecare bloc de 8×8 (se folosesc tabelele din segmentul DQT);
 - 5) Transformarea inversă a DCT (IDCT) pentru fiecare bloc de 8×8
 - 6) Reeșantionarea. La acest punct o imagine de pixeli de dimensiunea originală a fost reconstruită.
 - 7) Refacerea datelor în format RGB din YUV. Acest pas nu se aplică pentru imaginile pe nivele de gri.
- Pașii 2 – 7 trebuie repetați pentru toate blocurile de 8×8, până când toate blocurile sunt prelucrate.

Fișierele JPEG

- JPEG a dezvoltat un **format de fișier** care descrie modul de organizare a conținutului/șirului de biți codat – pentru a avea atât **parametrii de codare**, cât și **imaginea**
- Astfel, un fișier JPEG este împărțit în mai multe segmente
 - cu ajutorul unor cuvinte de cod - delimitatori (fiecare delimitator este precedat de 1 octet 0xFF).
 - aceste segmente sunt utilizate pentru păstrarea informațiilor
 - generale despre imagine,
 - despre modul de compresie a imaginii (rata de compresie, tabele de cuantizare, tabele Huffman, etc.),
 - precum și datele ce descriu conținutul imaginii.
- acestea permit implementarea JPEG pe diferite platforme, iar decompresia poate fi realizată fără alte probleme.
- au de obicei extensia „.jpeg” sau „.jpg” dar se mai utilizează ocazional și extensia „.jfif” (JPEG File Interchange Format).

Delimitatori de segmente – **JPEG**

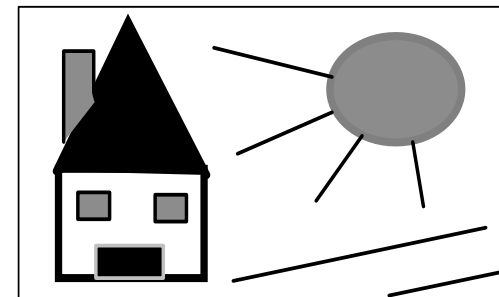
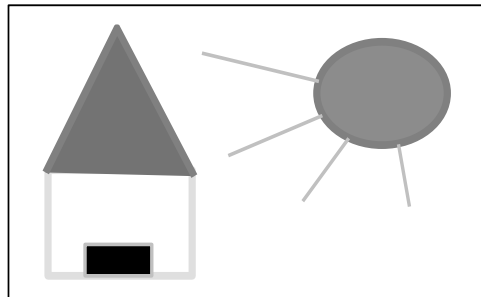
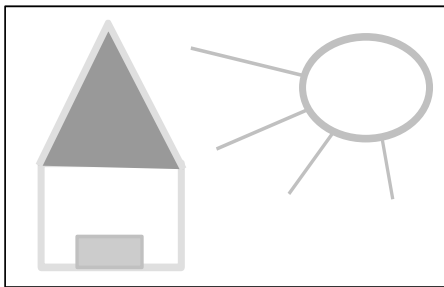
| Nume delimitator | Identificator delimitator | Descriere |
|------------------|---------------------------|---------------------------|
| SOI | 0xD8 | Start imagine |
| APP0 | 0xE0 | Segmentul aplicației JFIF |
| APPn | 0xE1 – 0xEF | Alte segmente APP |
| COM | 0xFE | Comentariu text |
| DQT | 0xDB | Tabelul de cuantizare |
| SOF0 | 0xC0 | Start cadru |
| DHT | 0xC4 | Tabelul Huffman |
| SOS | 0xDA | Start scanare |
| EOI | 0xD9 | Sfârșitul imaginii |

Segmente principale:

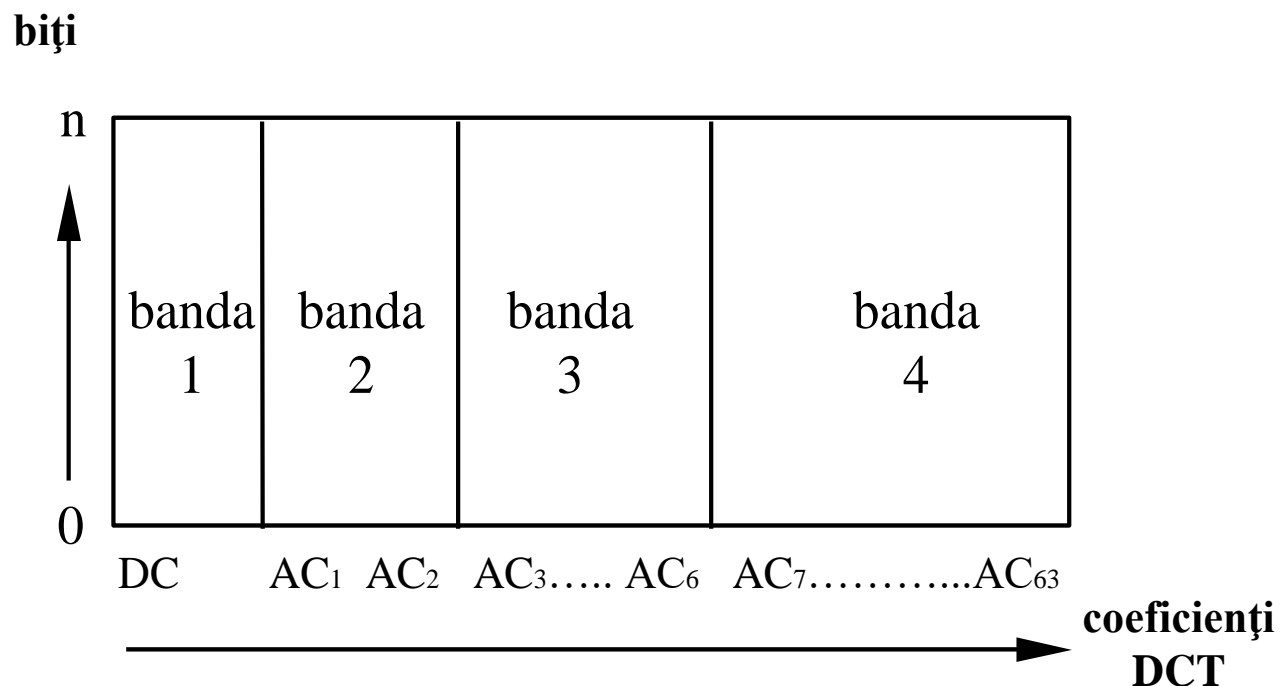
- Delimitatorul **SOI** - apare la începutul imaginii
- **APP0** - conține date pentru identificarea fișierului
- **APPn** (unde n poate lua valori de la 1 la 15) - conțin parametri de achiziție ai imaginii (metadata) – acest segment este opțional.
- **DQT** - unul sau mai multe tabele de cuantizare
- **SOF0** - pentru start cadru – indică faptul că modul de operare pentru compresie este secvențial bazat pe DCT (pentru celelalte moduri există alte delimitatoare) și specifică dimensiunea imaginii, numărul de componente și rata de sub eșantionare a componentelor
- **DHT** - Unul sau mai multe tabele Huffman
- **SOS** - specifică începutul scanării datelor ce descriu conținutul imaginii
- **EOI** - pentru sfârșitul imaginii

Modul de codare progresiv

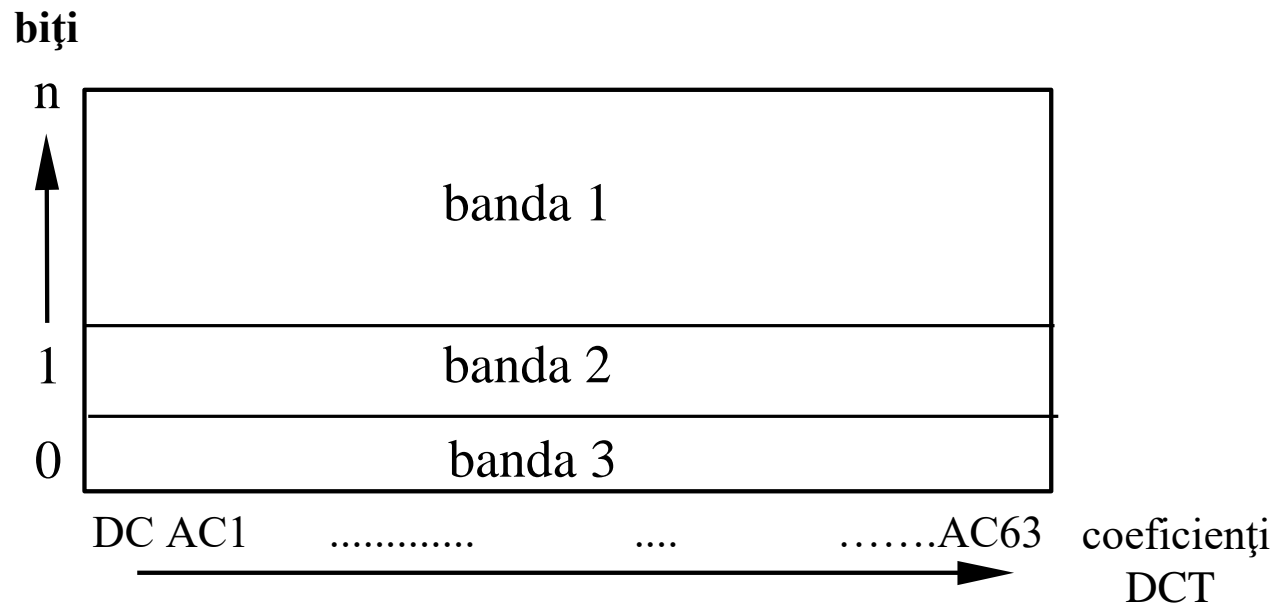
- diferite fata de modul de baza:
 - 8-12 biti pentru reprezentarea planurilor de culoare
 - codare Huffman sau aritmetica
- tipuri de algoritmi
 - algoritm progresiv cu selectie spectrala
 - algoritm progresiv cu aproximari succesive
 - algoritm progresiv combinat



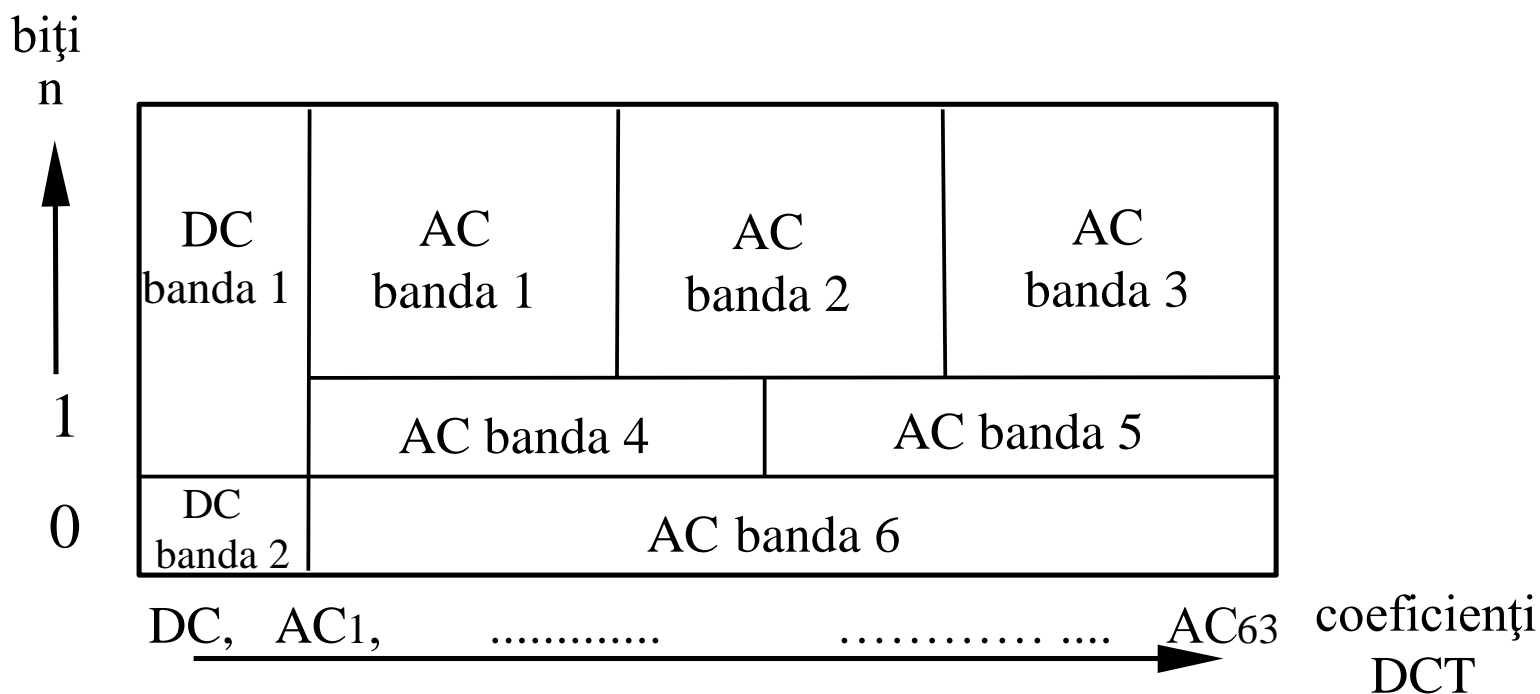
Algoritmul progresiv cu selectie spectrala



Algoritmul progresiv cu aproximari succesive



Algoritmul progresiv combinat



Etapele de decompresie progresiva



a) imagine la momentul t_1



b) imagine la momentul $t_2 > t_1$



c) imagine la momentul $t_3 > t_2$



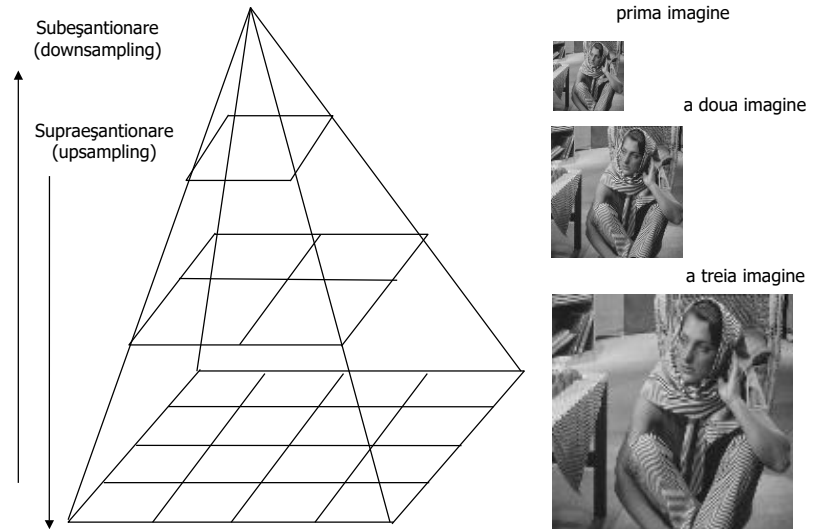
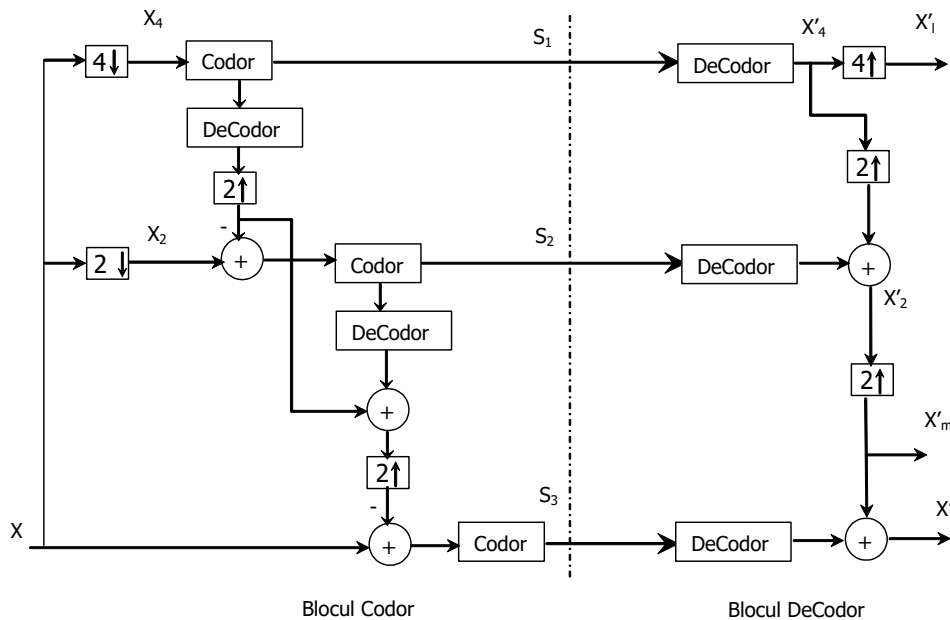
d) imagine originală ($t=t_4$)

Modul Ierarhic de operare compresie

- **modul ierarhic**
 - fiecare componenta a imaginii este codat ca o secvență de cadre:
 - primul cadru este o versiune de rezoluție joasă a imaginii originale (subeșantionată).
 - cadrele care urmează sunt cadre diferență între planurile imaginii sursă subeșantionate și planurile de referință reconstruite (supraeșantionate).
 - Cadrele pot fi codate folosind fie
 - JPEG cu pierderi
 - JPEG fără pierderi
 - combinat (ultima etapă JPEG fără pierderi)
 - Codarea ierarhică este folositoare atunci când avem nevoie de imagini la rezoluții multiple.
 - Exemplu: o aplicație poate afișa o imagine la rezoluție înaltă pe o stație grafică performantă și de asemenea poate afișa o imagine la rezoluție joasă pentru un simplu PC.

Codarea JPEG ierarhica

- Fiecare componenta a imaginii este codat ca o secvență de cadre:
 - primul cadru este o versiune de rezoluție joasă a imaginii originale (subeșantionată).
 - cadrele care urmează sunt cadre diferență între planurile imaginii sursă subeșantionate și planurile de referință reconstruite (supraeșantionate).



Quantization aspects

- An image at 100% quality has (almost) no loss, and 1% quality is a very low quality image. In general, quality levels of 90% or higher are considered "high quality", 80%-90% is "medium quality", and 70%-80% is low quality. Anything below 70% is typically a very low quality image.
- **Standard Quantization Tables**
 - These tables are referred to as "50%". The JPEG Standard also defines a scaling algorithm that can be used to alter these values to approximate the quality range from 1% to 100%. (This algorithm makes it easy for an application to provide 100 quality levels without hard-coding 100 pairs of quantization tables.) Many applications follow the JPEG Standard and use the sample quantization tables and scaling algorithm to quickly apply a selected quality level.
- **Non-Standard Quantization Tables**
 - The JPEG Standard's quantization tables are explicitly provided as an *example*; compliance is neither required nor essential. They are called the "Standard Quantization Tables" because they are described as examples in the JPEG Standard document, and not because of some standard requirement to use them.
 - Applications do not need to follow the JPEG Standard when determining quantization tables. Many devices and applications use their own, custom quantization tables. These are referred to as "non-standard" since they do not follow the examples in the JPEG Standard. Adobe, for example, defaults to using non-standard quantization tables and scaling algorithms. Similarly, if your digital camera has quality settings for "High", "Medium", and "Low", then it is actually referring to three hard-coded sets of quantization tables that are often non-standard.
- **JPEG %**
 - The JPEG % algorithm evaluates the values in the quantization tables. If the tables align with the JPEG Standard, then it identifies the required scaling factor. For non-standard tables, the algorithm estimates the percentage needed to achieve the same quality using the JPEG Standard values.
 - The results from JPEG % include:
 - The percentage needed to achieve the same quality using the JPEG Standard values.
 - Whether the percent value matches the JPEG Standard or is an estimate based on non-standard quantization tables.
 - The raw quantization tables extracted from the JPEG.
 - If the picture is not a JPEG, then JPEG % identifies the file as lossless and equivalent to 100% quality.

-
- The JPEG Standard ([CCITT/ITU T.81 Annex K](#) and [RFC 2435 section 4.2](#)) defines an approach that uses a scalar to adjust a set of well-defined quantization tables. However, the JPEG Standard uses two separate algorithms; one computation is used when scaling between 50% and 100% quality, and a different algorithm scales between 1% and 50% quality.
 - For JPEG types 0 and 1 (and their corresponding types 64 and 65), Q values between 1 and 99 inclusive are defined as follows. Other values less than 128 are reserved. Additional types are encouraged to use this definition if applicable.
 - Both type 0 and type 1 JPEG require two quantization tables. These tables are calculated as follows. For $1 \leq Q \leq 99$, the Independent JPEG Group's formula [5] is used to produce a scale factor S as:

$$S = 5000 / Q \quad \text{for } 1 \leq Q \leq 50$$

$$S = 200 - 2 * Q \quad \text{for } 51 \leq Q \leq 99$$

$$Lq[i,j] = (\text{jpeg_luma_quantizer}[i,j] * s + 50) / 100;$$

- This value is then used to scale Tables K.1 and K.2 from [1] (saturating each value to 8 bits) to give quantization table numbers 0 and 1, respectively.

-
- **Adobe**
 - The JPEG Standard defines one approach for determining quantization tables, but it is not the only approach. For example, Adobe Photoshop offers multiple scaling methods:
 - **Save As.** One method used by Photoshop is seen when using "Save As" and allows the user to select one of 12 quality levels with names like "Maximum", "High", and "Low".
 - **Save for Web.** Another method appears when using "Save for Web" and permits the user to select a quality value from 0 to 100. However, saving an image with Photoshop at "75" is not the same as saving a JPEG using the JPEG Standard algorithm at 75% quality.
 - **Advanced.** Photoshop includes an advanced option to save using the JPEG Standard algorithm rather than its own quantization tables. However, this option is buried in the menus and the location varies by software version.