

UNIVERSITATEA TEHNICĂ A MOLDOVEI  
Facultatea Calculatoare, Informatică și Microelectronică  
Departamentul Informatică și Ingineria Sistemelor

# **GRAFICA PE CALCULATOR**

## **TEMA 5. COMPRESIA DE IMAGINI**

l. u., dr. NASTAS Andrei

- 5.1. Reprezentarea numerică a imaginilor
- 5.2. Reprezentarea imaginii în format necompresat
- 5.3. Metode și abordări ale compresiei imaginii
- 5.4. Transformări folosite în compresia imaginilor
  - 5.4.1. Transformări ortogonale
  - 5.4.2. Transformări bi-dimensionale
  - 5.4.3. Transformarea Karhunen-Loève (KLT)
  - 5.4.4. Transformarea Walsh-Hadamard (WHT)
  - 5.4.5. Transformarea Haar
  - 5.4.6. Transformata Discretă Cosinus (DCT)
  - 5.4.7. Transformarea Discretă Sinus
- 5.5. Compresia JPEG (Joint Photographers Experts Group)
  - 5.5.1. Modul de bază (baseline)
  - 5.5.2. Modul de codare cu pierderi expandat
  - 5.5.3. Compresia secvențială JPEG fără pierderi
  - 5.5.4. Compresia JPEG ierarhică

## 5.4. Transformări folosite în compresia imaginilor

Conceptul matematic de transformare este important în multe domenii, printre care și cel al compresiei de imagini. O imagine poate fi compresată prin transformarea pixelilor săi (care sunt corelați) într-o reprezentare unde aceștia sunt decorelați. Compresia este obținută dacă valorile noi sunt mai mici, în medie, decât cele originale. Compresia cu pierdere de informație poate fi obținută prin cuantizarea valorilor transformate. Decodorul primește valorile transformate din secvența compresată și reconstruiește datele originale (exacte sau approximate), prin aplicarea transformării inverse. Transformările discutate în continuare sunt ortogonale.

Termenul de decorelare se referă la faptul că valorile transformate sunt independente unele de altele. Ca urmare, ele pot fi codate independent, ceea ce face mai simplă construirea unui model statistic. O imagine poate fi compresată, dacă reprezentarea sa are redundanță. Redundanța în imagini derivă din corelarea pixelilor. Dacă se transformă imaginea într-o reprezentare în care pixelii sunt decorelați, se elimină redundanța și imaginea a devenit în totalitate compresată. Se consideră cazul în care se scanează o imagine în ordinea rastrului și se grupează perechile de pixeli adiacenți. Deoarece pixelii sunt corelați, cei doi pixeli ai unei perechi, în mod normal, au valori similare.

## 5.4. Transformări folosite în compresia imaginilor

În continuare se consideră perechile de pixeli ca puncte în spațiul bi-dimensional, și se reprezintă grafic. Se știe că toate punctele de forma  $(x, x)$  sunt localizate pe prima bisectoare,  $y = x$ , așa că este de așteptat ca punctele considerate să fie concentrate în jurul acesteia. Figura 5.4, *a* arată rezultatele reprezentării pixelilor unei imagini oarecare, în care un pixel are valori de la 0 la 255. Cele mai multe puncte sunt grupate în jurul acestei linii, și doar câteva puncte sunt localizate departe de ea. Acum se transformă imaginea prin rotirea tuturor punctelor cu  $45^\circ$  în sensul acelor de ceasornic în jurul originii, așa încât prima bisectoare coincide acum cu axa  $x$ , ca în fig. 5.4, *b*.

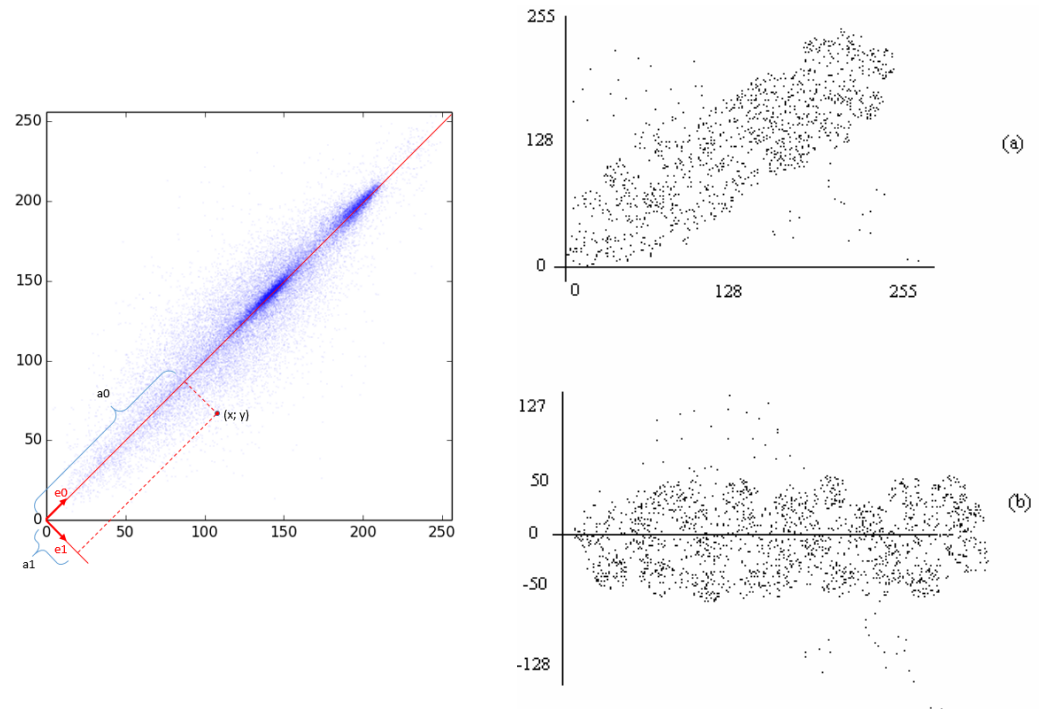


Fig. 5.4. Rotirea unui grup de puncte

## 5.4. Transformări folosite în compresia imaginilor

Aceasta se face prin intermediul transformării simple

$$(x^*, y^*) = (x, y) \begin{pmatrix} \cos 45^\circ & -\sin 45^\circ \\ \sin 45^\circ & \cos 45^\circ \end{pmatrix} = (x, y) \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix} = (x, y) \mathbf{R} \quad (5.2)$$

unde matricea de rotație  $R$  este ortonormală (adică produsul scalar al unui rând cu el însuși este 1, produsul scalar al rândurilor diferite este 0, și la fel pentru coloane). Transformarea inversă este

$$(x, y) = (x^*, y^*) \mathbf{R}^{-1} = (x^*, y^*) \mathbf{R}^T = (x^*, y^*) \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix} \quad (5.3)$$

(Inversa unei matrici ortonormale este transpusa sa).

Este evident că majoritatea punctelor ajung să aibă coordonatele  $y$  nule sau aproape nule, în timp ce coordonatele  $x$  nu se modifică foarte mult. Distribuțiile coordonatelor  $x$  și  $y$  (adică pixelii cu număr impar și par dintr-o imagine) înainte de rotație nu diferă cu mult, pe când, după rotație, distribuția coordonatelor  $x$  rămâne aproape la fel, dar cea a coordonatelor  $y$  este concentrată în jurul lui zero.

## 5.4. Transformări folosite în compresia imaginilor

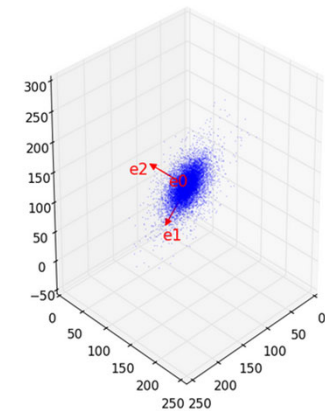
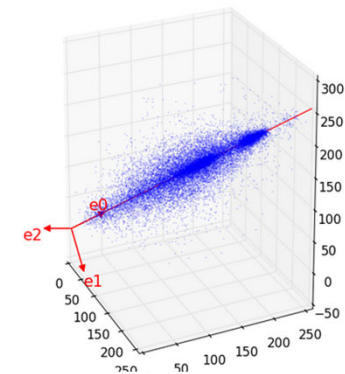
Cum coordonatele punctelor sunt cunoscute înainte și după rotație, este simplu să se măsoare reducerea ce intervine în corelația punctelor, prin calculul funcției de corelație  $\sum_i x_i y_i$  dintre puncte. În acest caz, compresia fără pierdere a imaginii poate fi efectuată prin simpla folosire a pixelilor transformați în secvența compresată. Dacă se acceptă compresie cu pierdere de informație, atunci toți pixelii pot fi cuantizați, obținându-se numere și mai mici. De asemenea, se pot separa toți pixelii cu număr impar (cei care creează coordonatele  $x$  ale perechilor), urmați apoi de toți pixelii cu număr par. Aceste două secvențe sunt numite *vectorii coeficienților* transformării. A doua secvență constă din numere mici și poate avea, după cuantizare, șiruri de zerouri, care pot conduce la o compresie și mai bună.

Se poate arăta că dispersia totală a pixelilor nu se modifică prin rotație, din moment ce matricea de rotație este ortonormală. Totuși, deoarece dispersia noilor coordonate  $y$  este mică, cea mai mare parte din dispersie este acum concentrată în coordonatele  $x$ . Dispersia este uneori numită *energia* distribuției pixelilor, astfel încât se poate afirma că rotația a concentrat (sau compactat) energia în coordonata  $x$  și a realizat compresia în acest fel.

## 5.4. Transformări folosite în compresia imaginilor

Această transformare simplă poate fi extinsă cu ușurință la orice număr de dimensiuni. În loc de a selecta perechi de pixeli adiacenți, se pot selecta triplete. Fiecare triplet devine un punct în spațiul tri-dimensional, și aceste puncte formează o regiune concentrată în jurul liniei care formează unghiuri de  $45^\circ$  cu cele trei axe de coordonate. Când această linie este rotită astfel încât să coincidă cu axa x, coordonatele y și z ale punctelor transformate devin numere mici. Transformarea este făcută prin multiplicarea fiecărui punct cu o matrice de rotație de dimensiune  $3 \times 3$ , ortonormală. Punctele transformate sunt apoi separate în trei vectori de coeficienți, din care ultimii doi sunt alcătuiți din numere mici. Pentru compresie maximă, fiecare vector de coeficienți trebuie cuantizat separat.

Această idee se poate extinde la mai mult de trei dimensiuni, cu singura diferență că nu se pot vizualiza spații de dimensiuni mai mari de trei. Unele metode de compresie, cum ar fi JPEG, împart o imagine în blocuri de  $8 \times 8$  pixeli fiecare, și rotesc fiecare bloc de două ori. Această rotație dublă produce un set de 64 de valori transformate, din care prima, numită „coeficient DC” sau de curent continuu, este mare, și celelalte 63, numite „coeficienții AC” sau de curent alternativ, sunt, de obicei, mici. Astfel, această transformare concentrează energia în prima din cele 64 de dimensiuni.



## 5.4.1. Transformări ortogonale

Transformările de imagine folosite în practică trebuie să fie rapide și, de preferință, simplu de implementat. Aceasta sugerează folosirea transformărilor liniare. Într-o astfel de transformare, fiecare valoare transformată  $c_i$  este o sumă ponderată a pixelilor  $d_j$ , unde fiecare este multiplicat cu un factor (sau coeficient de transformare)  $w_{ij}$ . Astfel,

$$c_i = \sum_j d_j w_{ij}, \text{ pentru } i, j = 1, 2, \dots, n. \quad (5.5)$$

Pentru  $n = 4$ , relația precedentă se scrie matriceal

$$\begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{pmatrix} = \begin{pmatrix} w_{11} & w_{12} & w_{13} & w_{14} \\ w_{21} & w_{22} & w_{23} & w_{24} \\ w_{31} & w_{32} & w_{33} & w_{34} \\ w_{41} & w_{42} & w_{43} & w_{44} \end{pmatrix} \begin{pmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \end{pmatrix}$$

În general

$$C = W \cdot D. \quad (5.5')$$

Fiecare rând al lui  $W$  este numit *vector al bazei*.



## 5.4.1. Transformări ortogonale

Se dorește determinarea valorilor ponderilor  $w_{ij}$ , astfel încât prima valoare transformată  $c_1$  să fie mare, și restul valorilor  $c_2, c_3, \dots$  să fie mici. Din relația  $c_i = \sum_j d_j w_{ij}$ , se observă că  $c_i$  va fi mare, când fiecare pondere  $w_{ij}$  consolidează contribuția lui  $d_j$  la  $c_i$ . Aceasta are loc, de exemplu, când vectorii  $w_{ij}$  și  $d_j$  au valori și semne similare. Invers,  $c_i$  va fi mic, dacă toate ponderile  $w_{ij}$  sunt mici și jumătate din ele au semnul opus lui  $d_j$ . Astfel, când se obține un  $c_i$  mare, înseamnă că vectorul  $w_{ij}$  al bazei este similar cu vectorul de date  $d_j$ . Un  $c_i$  mic, pe de altă parte, înseamnă că  $w_{ij}$  și  $d_j$  au forme diferite. În concluzie, vectorii bazei  $w_{ij}$  pot fi interpretați ca mijloace de a extrage caracteristici din vectorul de date.

În practică, ponderile ar trebui să fie independente de date. Altfel, ponderile ar trebui incluse în secvența compresată, pentru a fi folosite de decodor. Acest lucru, combinat cu faptul că datele sunt valorile pixelilor, care sunt nenegative, sugerează o modalitate de a alege vectorii bazei. Primul vector, cel care produce  $c_1$ , ar trebui să fie alcătuit din valori pozitive, poate chiar identice. Aceasta va întări valorile nenegative ale pixelilor. Fiecare din ceilalți vectori ar trebui să aibă jumătate din elemente pozitive, cealaltă jumătate, negative. Când sunt multiplicați cu valorile nenegative ale datelor, astfel de vectori tind să producă o valoare mică. O alegere bună a vectorilor bazei ar fi dacă aceștia ar fi foarte diferiți unul de altul, și astfel pot extrage mai multe caracteristici. Aceasta duce la ideea ca vectorii bazei să fie ortogonali. Dacă matricea de transformare  $W$  este ortogonală, transformarea în sine se numește ortogonală. O altă observație care ajută la selectarea vectorilor bazei este că aceștia ar trebui să aibă frecvențe din ce în ce mai mari, extrăgând astfel caracteristici de frecvență mai înaltă din date pe parcursul calculului valorilor transformate.

## 5.4.1. Transformări ortogonale

Aceste considerații sunt satisfăcute de matricea ortogonală:

$$W = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{pmatrix} \quad (5.6)$$

Primul vector al bazei (rândul de sus al lui  $W$ ) constă numai din valori 1, deci frecvența sa este zero. Fiecare din vectorii următori are doi de +1 și doi de -1, deci produc valori transformate mici, și frecvențele lor (măsurate ca numărul de schimbări de semn de-a lungul vectorului bazei) devin mai înalte. Această matrice este similară cu transformarea Walsh-Hadamard.

## 5.4.3. Transformarea Karhunen-Loève (KLT)

Transformarea Karhunen-Loève (KLT) are cea mai bună eficiență din punct de vedere al compactării energiei (sau, altfel spus, decorelare a pixelilor), dar are mai mult o valoare teoretică decât una practică.

Se consideră o imagine care se împarte în  $k$  blocuri de câte  $n$  pixeli fiecare, unde  $n$  este de obicei 64, dar poate avea și alte valori, și  $k$  depinde de mărimea imaginii. Se consideră blocurile de vectori  $b^{(i)}$ , unde  $i = 1, 2, \dots, k$ . Pe baza acestora se calculează vectorul medie  $\bar{b} = (\sum_i b^{(i)})/k$ . Se definește un nou set de vectori  $v^{(i)} = b^{(i)} - \bar{b}$ , care vor avea media zero. Matricea transformării KLT are dimensiunea  $n \times n$  și se notează cu  $A$ . Rezultatul transformării vectorului  $v^{(i)}$  este vectorul pondere  $w^{(i)} = A \cdot v^{(i)}$ . Valoarea medie a lui  $w^{(i)}$  este de asemenea zero. Se construiește o matrice  $V$  ale cărei coloane sunt vectorii  $v^{(i)}$  și o altă matrice ale cărei coloane sunt vectorii pondere  $w^{(i)}$ .

$$V = (v^{(1)}, v^{(2)}, \dots, v^{(k)}), W = (w^{(1)}, w^{(2)}, \dots, w^{(k)}) \quad (5.7)$$

Matricele  $V$  și  $W$  au fiecare  $n$  linii și  $k$  coloane. Din definiția lui  $w^{(i)}$  rezultă că  $W = A \cdot V$ .

Cei  $n$  vectori de coeficienți  $c^{(j)}$  din transformarea Karhunen-Loève sunt dați de relația:

$$c^{(j)} = (w^{(1)}, w^{(2)}, \dots, w^{(k)}), j = 1, 2, \dots, n. \quad (5.8)$$

Astfel, vectorul  $c^{(j)}$  este format din elementele "j" ale tuturor vectorilor pondere  $w^{(i)}$  cu  $i = 1, 2, \dots, k$  ( $c^{(j)}$  este coordonata  $j$  a vectorilor  $w^{(i)}$ ).

## 5.4.3. Transformarea Karhunen-Loève (KLT)

Se examinează elementele matricei produs ( $W \cdot W^T$ ) (aceasta este o matrice de dimensiuni  $n \times n$ ). Un element oarecare, din linia  $a$  și coloana  $b$ , al acestei matrice este o sumă de produse:

$$(W \cdot W^T)_{ab} = \sum_{i=1}^k w_a^{(i)} w_b^{(i)} = \sum_{i=1}^k c_i^{(a)} c_i^{(b)} = \mathbf{c}^{(a)} \mathbf{c}^{(b)}, \text{ pentru } a, b \in [1, n] \quad (5.9)$$

Deoarece valoarea medie a fiecărui vector  $w^{(i)}$  este zero, un element  $(W \cdot W^T)_{jj}$ , de pe diagonala principală a matricei produs, este varianța sau dispersia elementului  $j$  (sau a coordonatei  $j$ ) a vectorului  $w^{(i)}$ .

Elementele din afara diagonalei sunt covarianțele vectorilor  $w^{(i)}$ , adică un element  $(W \cdot W^T)_{ab}$  este covarianța coordonatelor  $a$  și  $b$  ale vectorilor  $w^{(i)}$ , care este egală cu produsul scalar  $c^{(a)} c^{(b)}$ . Un deziderat major al transformărilor aplicate imaginii este de a decorela coordonatele vectorilor. Din teoria probabilităților se știe că două coordonate sunt decorelate, dacă covarianța lor este zero. Un alt deziderat este acela de compactare a energiei, care, de fapt este în strânsă legătură cu primul. Având în vedere aceste lucruri, se urmărește găsirea unei matrice de transformare  $A$ , astfel încât produsul  $(W \cdot W^T)$  să fie o matrice diagonală.

Din definiția matricei  $W$  se obține:

$$W \cdot W^T = (AV) \cdot (AV)^T = A(V \cdot V^T)A^T \quad (5.10)$$

## 5.4.3. Transformarea Karhunen-Loève (KLT)

Matricea  $V \cdot V^T$  este simetrică, și elementele ei sunt covarianțele coordonatelor vectorilor  $v^{(i)}$ :

$$(V \times V^T)_{ab} = \sum_{i=1}^k v_a^{(i)} v_b^{(i)}, \text{ pentru } a, b \in [1, n] \quad (5.11)$$

Deoarece matricea  $(V \times V^T)$  este simetrică, vectorii săi proprii sunt ortogonali. Se normalizează acești vectori, pentru a fi ortonormali și se aleg ca linii ale matricei  $A$ . Rezultatul obținut este:

$$W \times W^T = A(V \times V^T)A^T = \begin{pmatrix} \lambda_1 & 0 & 0 & \dots & 0 \\ 0 & \lambda_2 & 0 & \dots & 0 \\ 0 & 0 & \lambda_3 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & \lambda_n \end{pmatrix} \quad (5.12)$$

Această alegere a matricei  $A$  conduce la o matrice  $W \times W^T$  diagonală, ale cărei elemente de pe diagonala principală sunt valorile proprii ale matricei  $V \times V^T$ . Matricea  $A$  este matricea transformării Karhunen – Loeve, liniile sale fiind vectorii bazei KLT și energiile (varianțele) vectorilor transformați sunt valorile proprii  $\lambda_1, \lambda_2, \dots, \lambda_n$  ai matricei  $(V \times V^T)$ . Vectorii bazei transformării KL nu sunt ficși, ei sunt dependenți de date, fiind calculați pe baza pixelilor imaginii originale. Din acest motiv, ei trebuie să incluși în secvența de date compresate, fapt ce scade eficiența acestei transformate.

## 5.4.4. Transformarea Walsh-Hadamard (WHT)

Această transformare are eficiență de compresie scăzută, nefiind folosită mult în practică. Cu toate acestea, este rapidă, deoarece poate fi calculată doar prin adunări, scăderi, și uneori, câte o deplasare la dreapta (pentru a împărți eficient printr-o putere a lui 2).

Dat fiind un bloc de  $N \times N$  de pixeli  $P_{xy}$  (unde  $N$  trebuie să fie o putere a lui 2,  $N = 2^n$ ), WHT bi-dimensională corespunzătoare și inversa sa sunt definite prin ecuațiile 5.16 și, respectiv 5.17:

$$\begin{aligned}
 H(u, v) &= \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} P_{xy} g(x, y, u, v) = \\
 &= \frac{1}{N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} P_{xy} (-1)^{\sum_{i=0}^{n-1} [b_i(x)p_i(x) + b_i(y)p_i(y)]}
 \end{aligned}
 \tag{5.16}$$

$$\begin{aligned}
 P_{xy} &= \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} H(u, v) h(x, y, u, v) = \\
 &= \frac{1}{N} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} H(u, v) (-1)^{\sum_{i=0}^{n-1} [b_i(x)p_i(x) + b_i(y)p_i(y)]}
 \end{aligned}
 \tag{5.17}$$

unde  $H(u, v)$  sunt rezultatele transformării (adică, coeficienții WHT), cantitatea  $b_i(u)$  este bitul  $i$  al reprezentării binare a numărului întreg  $u$ , iar  $p_i(u)$  este definit în funcție de  $b_i(u)$  prin ecuația 5.18.

$$\begin{aligned}
 p_0(u) &= b_{n-1}(u), \\
 p_1(u) &= b_{n-1}(u) + b_{n-2}(u), \\
 p_2(u) &= b_{n-2}(u) + b_{n-3}(u), \\
 &\dots\dots\dots \\
 p_{n-1}(u) &= b_1(u) + b_0(u).
 \end{aligned}
 \tag{5.18}$$

## 5.4.4. Transformarea Walsh-Hadamard (WHT)

De exemplu, se consideră  $u = 6 = 110_2$ . Biții zero, unu, și doi ai lui 6 sunt, respectiv, 0; 1 și 1, deci  $b_0(6) = 0$ ,  $b_1(6) = 1$ , și  $b_2(6) = 1$ .

Mărimile  $g(x, y, u, v)$  și  $h(x, y, u, v)$  sunt numite nuclee (kernels) (sau imagini de bază) ale WHT. Aceste matrice sunt identice. Elementele lor sunt doar +1 și -1, și sunt multiplicare prin factorul  $1/N$ . Ca urmare, transformarea WHT constă în multiplicarea fiecărui pixel din imagine cu +1 sau -1, adunare, și împărțirea sumei prin  $N$ . Deoarece  $N = 2^n$ , împărțirea prin  $2^n$  poate fi făcută prin deplasarea cu  $n$  poziții spre dreapta.

Nucleele WHT sunt prezentate, în formă grafică, pentru  $N = 4$ , în figura 5.5, unde albul reprezintă +1 și negrul -1 (factorul  $1/N$  este ignorat).

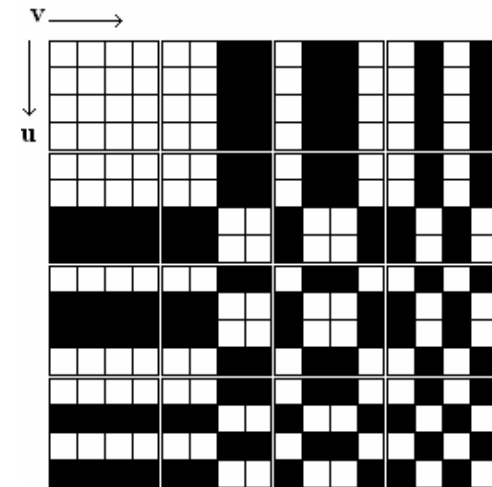


Fig. 5.5. Nucleul ordonat al WHT pentru  $N = 4$

## 5.4.4. Transformarea Walsh-Hadamard (WHT)

Rândurile și coloanele de blocuri din această figură corespund unor valori ale lui  $u$  și respectiv,  $v$ , de la 0 la 3. Numărul de schimbări de semn pe parcursul unui rând sau al unei coloane a unei matrice se numește *secvențierea* rândului sau coloanei.

Matricele corespunzătoare transformatei DWHT pot fi obținute recursiv și ca rearanjări ale matricelor discrete Hadamard care sunt de o importanță particulară în teoria codării. O matrice Hadamard de dimensiune  $N$  are proprietatea că  $HH^T = NI$  unde  $I$  este matricea unitate.

Matricele Hadamard ale căror dimensiuni sunt puteri ale lui doi pot fi construite astfel:

$$H_{2N} = \begin{bmatrix} H_N & H_N \\ H_N & -H_N \end{bmatrix} \quad (5.19)$$

cu  $H_1 = [1]$ . Astfel se obțin

$$H_2 = \begin{bmatrix} H_1 & H_1 \\ H_1 & -H_1 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (5.20)$$

$$H_4 = \begin{bmatrix} H_2 & H_2 \\ H_2 & -H_2 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \quad (5.21)$$

$$H_8 = \begin{bmatrix} H_4 & H_4 \\ H_4 & -H_4 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{bmatrix} \quad (5.22)$$



## 5.4.4. Transformarea Walsh-Hadamard (WHT)

Matricea  $H$  a transformării DWHT poate fi obținută din matricea Hadamard prin multiplicarea acesteia cu un factor de normalizare, astfel încât  $HH^T = I$ , și prin reordonarea liniilor în ordine descrescătoare a frecvenței. Normalizarea implică multiplicarea matricei cu  $\frac{1}{\sqrt{N}}$ . Reordonând  $H_8$ , se obține

$$H = \frac{1}{\sqrt{8}} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \end{bmatrix} \quad (5.23)$$

Deoarece matricea fără factorii de scalare constă din  $\pm 1$ , operația de transformare constă doar în adunare și scădere. Din acest motiv transformata este utilă în situații în care minimizarea câștigului de calcul este foarte importantă.

## 5.4.5. Transformarea Haar

Transformarea Haar este bazată pe funcția Haar  $h_k(x)$ , care este definită pentru  $x \in [0, 1]$  și pentru  $k = 0, 1, \dots, N - 1$ , unde  $N = 2^n$ .

Orice întreg  $k$  poate fi exprimat sub formă de sumă astfel  $k = 2^p + q - 1$ , unde  $0 \leq p \leq n - 1$ ,  $q = 0$  sau  $1$  pentru  $p = 0$ , și  $1 \leq q \leq 2^p$  pentru  $p \neq 0$ . De exemplu, pentru  $N = 4 = 2^2$ , se obține  $0 = 2^0 + 0 - 1$ ,  $1 = 2^0 + 1 - 1$ ,  $2 = 2^1 + 1 - 1$ , și  $3 = 2^1 + 2 - 1$ .

Funcția de bază a transformatei Haar este definită astfel:  $h_0(x) = h_{00}(x) = \frac{1}{\sqrt{N}}$ , pentru  $0 \leq x \leq 1$  (5.24)

$$\text{și } h_k(x) \stackrel{\text{def}}{=} h_{pq}(x) = \frac{1}{\sqrt{N}} \begin{cases} 2^{p/2}, \frac{q-1}{2^p} \leq x < \frac{(q-1)/2}{2^p} \\ -2^{p/2}, \frac{(q-1)/2}{2^p} \leq x < \frac{q}{2^p} \\ 0, \text{ în rest} \end{cases} \quad (5.25)$$

Matricea transformatei Haar,  $A_N$  de ordinul  $N \times N$  poate fi construită astfel: un element oarecare al matricei  $(i, j)$  este funcția de bază  $h_i(j)$ , unde  $i = 0, 1, \dots, N - 1$  și  $j = 0/N, 1/N, \dots, (N - 1)/N$  ( $i = 1$  implică  $p = 0$  și  $q = 1$ ). De exemplu:

$$A_2 = \begin{pmatrix} h_0(0/2) & h_0(1/2) \\ h_1(0/2) & h_1(1/2) \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \quad (5.26)$$

## 5.4.6. Transformata Discretă Cosinus (DCT)

Transformările cosinus și sinus discrete, (DCT, DST) aparțin familiei transformărilor trigonometrice cu aplicații în compresia/decompresia datelor. Dintre acestea, DCT este de departe mai folosită în practică, datorită proprietății de compactare a energiei.

- **DCT unidimensională**

În practică se folosește DCT bi-dimensională, dar pentru ușurința înțelegerii se consideră mai întâi DCT uni-dimensională. Se consideră formele de undă  $w(f) = \cos(f\vartheta)$ , pentru  $0 \leq \vartheta \leq \pi$ , cu frecvențele  $f = 0, 1, \dots, 7$ , și

$$\theta = \frac{\pi}{16}, \frac{3\pi}{16}, \frac{5\pi}{16}, \frac{7\pi}{16}, \frac{9\pi}{16}, \frac{11\pi}{16}, \frac{13\pi}{16}, \frac{15\pi}{16} \quad (5.27)$$

Fiecare formă de undă  $w(f)$  este eșantionată în opt puncte, pentru a forma un vector al bazei  $v_f$ . Cei opt vectori rezultați  $v_f$ ,  $f = 0, 1, \dots, 7$  (un total de 64 de numere) sunt prezentați în tabelul 5.1. Aceștia reprezintă baza pentru DCT uni-dimensională. Se observă similaritatea dintre acest tabel și matricea  $W$  din ecuația (5.5).

Tabelul 5.1. Vectori rezultați  $v_f$

$\theta$	0,196	0,589	0,982	1,374	1,767	2,160	2,553	2,945
$\cos 0\theta$	1	1	1	1	1	1	1	1
$\cos 1\theta$	0,981	0,831	0,556	0,195	-0,195	-0,556	-0,831	-0,981
$\cos 2\theta$	0,924	0,383	-0,383	-0,924	-0,924	-0,383	0,383	0,924
$\cos 3\theta$	0,831	-0,195	-0,981	-0,556	0,556	0,981	0,195	-0,831
$\cos 4\theta$	0,707	-0,707	-0,707	0,707	0,707	-0,707	-0,707	0,707
$\cos 5\theta$	0,556	-0,981	0,195	0,831	-0,831	-0,195	0,981	-0,556
$\cos 6\theta$	0,383	-0,924	0,924	0,383	-0,383	0,924	-0,924	0,383
$\cos 7\theta$	0,195	-0,556	0,831	0,981	0,981	-0,831	0,556	-0,195

## 5.4.6. Transformata Discretă Cosinus (DCT)

Acești opt vectori  $v_i$  sunt ortonormali (datorită alegerii particulare a celor opt puncte de eșantionare) și pot fi organizați într-o matrice de transformare  $8 \times 8$ . Pentru că această matrice este ortonormală, ea este o matrice de rotație, deci, DCT uni-dimensională poate fi interpretată ca o rotație în opt dimensiuni.

O altă interpretare a DCT uni-dimensională este aceea că se pot considera cei opt vectori ortonormali  $v_i$  ca bază a unui spațiu vectorial, și orice alt vector  $p$  poate fi exprimat în acest spațiu ca o combinație liniară a acestor  $v_i$ . De exemplu, se aleg ca date de test 8 numere corelate,  $p = (0,6; 0,5; 0,4; 0,5; 0,6; 0,5; 0,4; 0,55)$ . Se exprimăm vectorul  $p$  ca o combinație liniară a celor opt vectori ai bazei,  $p = \sum w_i v_i$ . Rezolvând acest sistem de opt ecuații se obțin cele opt ponderi:

$$w_0 = 0,506, w_1 = 0,0143, w_2 = 0,0115, w_3 = 0,0439, \\ w_4 = 0,0795, w_5 = -0,0432, w_6 = 0,00478, w_7 = -0,0077$$

## 5.4.6. Transformata Discretă Cosinus (DCT)

Ponderea  $w_0$  nu este cu mult diferită de elementele vectorului  $p$ , dar celelalte șapte ponderi sunt mult mai mici. Acest fapt indică modul în care DCT (sau orice altă transformare ortogonală) produce compresie. Cele 8 ponderi vor reprezenta pur și simplu elementele compresate ale vectorului  $p$ . Cuantizând cele opt ponderi, se poate crește considerabil compresia, în timp ce se pierde doar o cantitate mică de date.

Figura 5.6 ilustrează grafic această combinație liniară. Fiecare din cei opt vectori  $v_i$  este prezentat ca un rând de opt dreptunghiuri mici, gri, unde o valoare de +1 este colorată în alb, și -1 în negru. Fiecare din cele opt elemente ale vectorului  $p$  este exprimat ca suma ponderată a unei scări de gri cu opt nivele.

Cel mai simplu mod de a calcula DCT uni-dimensională, în practică, este cu relația

$$G_f = \frac{1}{2} C_f \sum_{t=0}^7 p_t \cos\left(\frac{(2t+1)f\pi}{16}\right) \quad (5.28)$$

unde

$$C_f = \begin{cases} \frac{1}{\sqrt{2}}, & f = 0, \\ 1, & f > 0, \end{cases} \text{ pentru } f = 0, 1, \dots, 7. \quad (5.29)$$

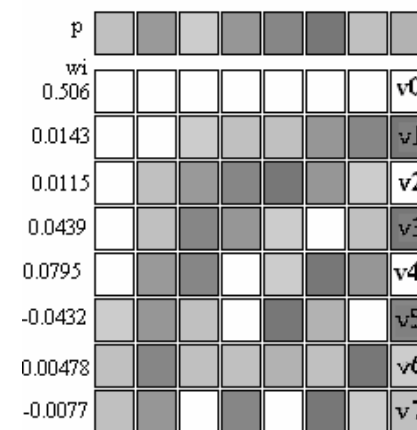


Fig. 5.6. Reprezentarea grafică a DCT uni-dimensional

## 5.4.6. Transformata Discretă Cosinus (DCT)

Se începe cu un set de opt valori de date  $p_t$  (pixeli, eşantioane de sunet, sau alte date) și se obține un set de opt coeficienți DCT,  $G_f$ . Decodorul primește coeficienții DCT în seturi de opt, și aplică transformarea inversă DCT (IDCT) pentru a reconstrui valorile de date originale (tot în grupuri de câte opt). IDCT se calculează cu relația

$$p_t = \frac{1}{2} \sum_{j=0}^7 C_j G_j \cos\left(\frac{(2t+1)j\pi}{16}\right), \quad \text{pentru } t=0, 1, \dots, 7 \quad (5.30)$$

### • DCT bi-dimensională

Din experiență se știe că pixelii unei imagini sunt corelați pe două dimensiuni, nu doar pe una (un pixel este corelat cu vecinii săi de la stânga și de la dreapta, deasupra și dedesubt). De aceea metodele de compresie a imaginii folosesc DCT bi-dimensională, dată de relația

$$G_{ij} = \frac{1}{\sqrt{2n}} C_i C_j \sum_{x=0}^{n-1} \sum_{y=0}^{n-1} p_{xy} \cos\left(\frac{(2y+1)j\pi}{2n}\right) \cos\left(\frac{(2x+1)i\pi}{2n}\right), \quad (5.31)$$

pentru  $0 \leq i, j \leq n - 1$ . Imaginea este împărțită în blocuri de  $n \times n$  pixeli  $p_{xy}$  (de obicei se folosește  $n = 8$ ), și ecuația (5.31) este folosită pentru a obține un bloc de  $8 \times 8$  coeficienți DCT,  $G_{ij}$ , pentru fiecare bloc de pixeli. Dacă compresia este cu pierdere de informație, coeficienții sunt cuantizați. Decodorul reconstruiește un bloc de valori de date (aproximate sau precise) prin calculul IDCT.

$$p_{xy} = \frac{1}{\sqrt{2n}} \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} C_i C_j G_{ij} \cos\left(\frac{(2y+1)j\pi}{2n}\right) \cos\left(\frac{(2x+1)i\pi}{2n}\right) \quad (5.32) \quad \text{unde } C_f = \begin{cases} \frac{1}{\sqrt{2}}, & f = 0, \\ 1, & f > 0, \end{cases} \quad \text{pentru } f = 0, 1, \dots, 7$$

## 5.4.6. Transformata Discretă Cosinus (DCT)

DCT bi-dimensională poate fi interpretată în două moduri diferite, ca o rotație (de fapt, două rotații separate), și ca bază a unui spațiu vectorial  $n$  - dimensional. În prima interpretare se consideră un bloc de  $n \times n$  pixeli. Mai întâi se consideră fiecare rând al acestui bloc ca un punct  $(p_{x,0}; p_{x,1}; \dots; p_{x,n-1})$  în spațiul  $n$ -dimensional, și se rotește punctul cu ajutorul transformării date de suma din interior

$$G1_{x,j} = C_j \sum_{y=0}^{n-1} p_{xy} \cos\left(\frac{(2y+1)j\pi}{2n}\right) \quad (5.33)$$

a ecuației (5.31). Aceasta transformare are ca rezultat un bloc  $G1_{x,j}$  de  $n \times n$  coeficienți, unde primul element al fiecărui rând este dominant și restul elementelor sunt mici. Suma exterioară a ecuației (5.31) este

$$G_{ij} = \frac{1}{\sqrt{2n}} C_i \sum_{x=0}^{n-1} G1_{x,j} \cos\left(\frac{(2x+1)i\pi}{2n}\right) \quad (5.34)$$

## 5.4.6. Transformata Discretă Cosinus (DCT)

Aici, coloanele lui  $G_{1_{x,j}}$  sunt considerate puncte în spațiul  $n$ -dimensional, și sunt rotite. Rezultatul este un coeficient mare în colțul stânga-sus al blocului și  $n^2 - 1$  coeficienți mici în rest. Această interpretare consideră DCT bi-dimensional ca două rotații separate, fiecare în  $n$  dimensiuni. Este interesant de observat că două rotații în  $n$  dimensiuni sunt mai rapide decât una în  $n^2$  dimensiuni, deoarece în al doilea caz este necesară o matrice de rotație de dimensiune  $n^2 \times n^2$ .

A doua interpretare (presupunând  $n = 8$ ) folosește ecuația (5.31) pentru a crea 64 blocuri de  $8 \times 8$  valori fiecare. Cele 64 de blocuri sunt apoi folosite ca bază a unui spațiu de vectori 64-dimensional (sunt imagini de bază). Imaginile de bază folosite în DCT bi-dimensională sunt date în figura 5.7. Orice bloc  $B$  de  $8 \times 8$  pixeli poate fi exprimat ca o combinație liniară a imaginilor de bază, și cele 64 de ponderi ale acestei combinații liniare sunt coeficienții DCT ai blocului  $B$ .



## 5.4.6. Transformata Discretă Cosinus (DCT)

Compresia unei imagini cu DCT presupune parcurgerea următorilor pași:

- Se împarte imaginea în  $k$  blocuri  $B_i$ ,  $i=1,2,\dots,k$ , de  $n \times n$  (obișnuit,  $8 \times 8$ ) pixeli fiecare.
  - Se aplică DCT bi-dimensională fiecărui bloc  $B_i$ . Aceasta transformare exprimă blocul ca o combinație liniară a celor 64 de imagini de bază. Rezultatul este un bloc, numit vector  $W^{(i)}$  de 64 de ponderi  $w_j^{(i)}$ , unde  $j = 0, 1, \dots, 63$ .
  - Cei  $k$  vectori  $W^{(i)}$  ( $i = 1, 2, \dots, k$ ) sunt împărțiți în 64 de vectori de coeficienți  $C^{(j)}$ , unde cele  $k$  elemente ale vectorului  $C^{(j)}$  sunt  $(w_j^{(1)}, w_j^{(2)}, \dots, w_j^{(k)})$ . Primul vector de coeficienți  $C^{(0)}$  este format din cei  $k$  coeficienți DC.
  - Fiecare vector de coeficienți  $C^{(j)}$  este cuantizat separat pentru a produce un vector cuantizat  $Q^{(j)}$ , care reprezintă datele compresate.
- Decodorul citește cei 64 de vectori de coeficienți cuantizați  $Q^{(j)}$ , îi folosește pentru a construi  $k$  vectori de ponderi  $W^{(i)}$ , și aplică IDCT fiecărui vector de ponderi, pentru a reconstrui cei 64 de pixeli ai blocului  $B_i$ .

## 5.4.7. Transformarea Discretă Sinus

Transformarea discretă sinus, DST, este complementara DCT. DCT asigură performanțe apropiate transformatei K-L optime, în ceea ce privește compactarea, când corelația coeficienților  $\rho$  este mare, iar DST asigură performanțe apropiate transformatei K-L optime, când  $\rho$  este mic. Datorită acestei proprietăți, este adesea folosită ca transformată complementară a DCT în codarea de imagini și audio.

Elementele matricei transformate pentru o DST de dimensiune  $n \times n$  sunt date de

$$[S]_{i,j} = \sqrt{\frac{2}{n}} \sin \frac{\pi(i+1)(j+1)}{2n}; \quad i, j = 0, 1, \dots, n-1 \quad (5.35)$$

Pentru a justifica folosirea mult mai frecventă a DCT în defavoarea DST, în continuare se prezintă diferențele dintre funcțiile sinus și cosinus și de ce aceste diferențe duc la o transformare sinus discretă ineficientă.

Funcția sinus este o funcție impară, iar funcția cosinus, pară. Deși singura diferență dintre cele două funcții este faza (adică funcția cosinus este o versiune defazată a sinusului), această diferență este suficientă pentru a le inversa paritatea. Pentru a înțelege diferența dintre DCT și DST se examinează cazul uni-dimensional. DCT uni-dimensională, dată de ecuația (5.28), folosește funcția  $\cos((2t+1)f\pi/16)$  pentru  $f = 0, 1, \dots, 7$ . Pentru primul termen, unde  $f = 0$ , această funcție devine  $\cos(0)$ , care este 1. Acest termen este coeficientul DC, care produce media celor opt valori de date supuse transformării.

## 5.4.7. Transformarea Discretă Sinus

DST este bazată în mod similar pe funcția  $\sin((2t + 1) f\pi/16)$ , având ca rezultat un prim termen nul (din moment ce  $\sin(0) = 0$ ), care nu contribuie cu nimic la transformare, deci DST nu are un coeficient DC.

Dezavantajul acestui lucru poate fi observat când se consideră exemplul a opt valori de date identice ce trebuie transformate. Astfel de valori sunt, desigur, perfect corelate. Când sunt reprezentate grafic ele devin o linie orizontală. Aplicând DCT acestor valori, se produce doar un coeficient DC; toți coeficienții AC fiind nuli. DCT compactează toată energia datelor într-un unic coeficient DC, a cărui valoare este identică cu a datelor. IDCT poate reconstrui exact cele opt valori (cu excepția unor modificări minore date de precizia limitată de calcul). Aplicarea DST asupra aceluiași opt valori, pe de altă parte, conduce la șapte coeficienți AC a căror sumă este o formă de undă care trece prin cele opt puncte corespunzătoare datelor, dar oscilează între aceste puncte. Acest comportament, are trei dezavantaje, în principal:

1. Energia datelor originale nu este compactată;
2. Cei șapte coeficienți nu sunt decorelați (pe când datele sunt perfect corelate);
3. Cuantizând cei șapte coeficienți se poate ajunge la o puternică scădere a calității reconstrucției realizate de DST inversă.

## 5.5. Compresia JPEG (Joint Photographers Experts Group)

Domeniul compresiei (codării) de imagini este legat de minimizarea numărului de biți necesari pentru a reface o imagine, cu aplicații în special în transmisia și stocarea imaginilor. Aplicațiile din domeniul transmisiilor de imagini se întâlnesc în televiziunea radiodifuzată, comunicațiile spațiale, radar și sonar, rețele de telecomunicații, transmisii fax, teleconferințe etc. Compresia imaginilor este esențială din punct de vedere al memorării (stocării) imaginilor în aplicații de imagistică medicală, în tehnica video digitală, pentru realizarea documentelor multimedia etc. Noile tehnologii de compresie a imaginilor oferă o soluție posibilă pentru integrarea aplicațiilor de imagini și video digitale. Ratele de compresie au ajuns în prezent până la 1:100, depinzând de calitatea imaginii refăcute. Tehnica de compresie nu este suficientă pentru a putea rezolva problemele care apar în aplicațiile multimedia. Pentru a putea realiza portabilitatea aplicațiilor de imagini și secvențe video digitale pe mai multe sisteme, este necesară implementarea unor standarde pentru compresia datelor multimedia. Aceste standarde stabilesc modalitățile de stocare și transmisie a datelor compresate în vederea posibilității utilizării lor. Cel mai utilizat standard de compresie a imaginilor statice este standardul JPEG, creat de Joint Photographics Experts Group. Metoda de compresie este de tip "cu pierdere", fiind concepută astfel încât să se profite de limitările în percepția video a ochiului uman. Acest standard permite setarea raportului calitate/compresie și lucrează cu aceleași nivele de culoare, în număr de 24 (16,7 milioane de culori), indiferent de numărul total de culori din imagine. În momentul de față este unul dintre cele mai frecvent întâlnite formate de fișiere grafice.

# ÎNTREBĂRI