Lucrarea 8

Transformări unitare

BREVIAR TEORETIC

Transformările reprezintă o categorie de prelucrări ce include operaţii de tip integral, la calculul noii valori a unui pixel al imaginii transformate contribuind valorile tuturor pixelilor din imaginea originală.

Termenul de transformare se referă la o clasă de matrici unitare folosite pentru a reprezenta imagini. O matrice A de dimensiune $N \times N$ este unitară dacă inversa ei este matricea A^{*T}:

$$A \cdot A^{-1} = A \cdot A^{*T} = A^{*T} \cdot A = I_N \tag{8.1}$$

unde * reprezintă operaţia de complementare în mulţimea numerelor complexe, ^{T} reprezintă operaţia de transpunere a unei matrici, iar I_N este matricea identitate de dimensiune $N \times N$:

$$I_N = \begin{pmatrix} 1 & 0 & 0 & ... & 0 \\ 0 & 1 & 0 & ... & 0 \\ 0 & 0 & 1 & ... & 0 \\ . & . & . & ... & . \\ . & . & . & ... & . \\ 0 & 0 & 0 & ... & 1 \end{pmatrix} \tag{8.2}$$

Pentru un vector uni-dimensional $\underline{u}$ de dimensiune N, de forma:

$$\underline{u} = \begin{bmatrix} u(0) \\ u(1) \\ . \\ . \\ . \\ u(N-1) \end{bmatrix} = [u(0), u(1), ..., u(N-1)]^T \tag{8.3}$$

se numeşte **transformare unitară directă** relaţia:

$$v(k) = \sum_{n=0}^{N-1} a(k,n) \cdot u(n), \quad 0 \leq k \leq N-1 \tag{8.4}$$

unde $v(k)$ reprezintă elementele vectorului transformat $\underline{v}$, iar $a(k,n)$ sînt elementele matricii A. Matricial această relaţie se poate scrie astfel:

$$\underline{v} = A \cdot \underline{u} \tag{8.5}$$

Transormarea unitară inversă este dată de relaţia:

$$u(n) = \sum_{k=0}^{N-1} v(k) \cdot a^*(k,n), \quad 0 \leq n \leq N-1 \tag{8.6}$$

care se scrie matricial astfel:

$$\underline{u} = A^{*T} \cdot \underline{v} \tag{8.7}$$

Valorile vectorului $\underline{v}$ sunt o reprezentare a vectorului iniţial $\underline{u}$, într-un alt spaţiu. O astfel de reprezentare este utilă în filtrare, compresie, extragere de trăsături sau alte tipuri de analiză a imaginilor.

8.1 Transformări unitare bidimensionale

Pentru o imagine $u(m,n)$, de dimensiune $N \times N$, transformarea directă are următoarea formă:

$$v(k,l) = \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} u(m,n) \cdot a_{k,l}(m,n), \quad 0 \leq k,l \leq N-1 \tag{8.8}$$

iar transformarea inversă:

$$u(m,n) = \sum_{k=0}^{N-1} \sum_{k=0}^{N-1} v(k,l) \cdot a^*_{k,l}(m,n), \quad 0 \leq m,n \leq N-1 \tag{8.9}$$

unde coeficienţii $\{a_{k,l}(m,n)\}$ poartă numele de **transformare unitară bidimensională**, şi reprezintă un set de matrici de bază ortonormale, iar $v(k,l)$ reprezintă transformata imaginii $u(m,n)$.

Aceste matrici de bază respectă condiţia de **ortonormalitate**:

$$\sum_{m=0}^{N-1} \sum_{n=0}^{N-1} a_{k,l}(m,n) \cdot a^*_{k',l'}(m,n) = \delta(k-k', l-l') = \begin{cases} 1, & k = k' \;\&\; l = l', \\ 0, & \textit{în rest.} \end{cases} \tag{8.10}$$

pentru $\vee k, l$.

O transformare ca cea dată de relaţia (8.8) este caracterizată de N^4 coeficienţi $a_{k,l}(m,n)$. Pentru calculul unui singur element $v(k,l)$ (k şi l fixaţi) este nevoie de un număr de N^2 înmulţiri. Prin urmare complexitatea întregului calcul este $O(N^4)$. Complexitatea poate fi redusă la $O(N^3)$ dacă transformarea este separabilă.

O transformare este **separabilă**, dacă elementele $a_{k,l}(m,n)$ ce o definesc, pot fi scrise ca produs de alte două elemente, grupate după perechi de indici, astfel:

$$a_{k,l}(m,n) = a_k(m) \cdot b_l(n) = a(k,m) \cdot b(l,n) \tag{8.11}$$

unde, matricile $A = \{a(k,m)\}$ şi $B = \{b(l,n)\}$ trebuie să fie la rândul lor unitare, adică:

$$A \cdot A^{*T} = A^{*T} \cdot A = I_N \tag{8.12}$$

$$B \cdot B^{*T} = B^{*T} \cdot B = I_N \tag{8.13}$$

Dacă transformarea este separabilă, atunci relaţiile (8.8) şi (8.9) devin:

$$v(k,l) = \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} a(k,m) \cdot u(m,n) \cdot b(l,n) \tag{8.14}$$

$$u(m,n) = \sum_{k=0}^{N-1} \sum_{k=0}^{N-1} a^*(k,m) \cdot v(k,l) \cdot b^*(l,n) \tag{8.15}$$

care pot fi scrise matricial astfel:

$$V = A \cdot U \cdot B^T \tag{8.16}$$

$$U = A^{*T} \cdot V \cdot B^* \tag{8.17}$$

unde $U = \{u(m,n)\}$ reprezintă imaginea originală, iar $V = \{v(k,l)\}$ reprezintă imaginea transformată.

În practică se folosesc numai transformări separabile, pentru care, în plus, se alege $B = A$. În acest caz, vom avea o singură matrice A, unitară, care defineşte transfomarea, iar relaţiile (8.14) şi (8.15) devin:

$$v(k,l) = \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} a(k,m) \cdot u(m,n) \cdot a(l,n) \tag{8.18}$$

$$u(m,n) = \sum_{k=0}^{N-1} \sum_{k=0}^{N-1} a^*(k,m) \cdot v(k,l) \cdot a^*(l,n) \tag{8.19}$$

Matricial, relaţiile (8.16) şi (8.17) se scriu după cum urmează:

$$V = A \cdot U \cdot A^T \tag{8.20}$$

$$U = A^{*T} \cdot V \cdot A^* \tag{8.21}$$

8.1.1 Proprietăţile transformărilor unitare

În continuare vor fi prezentate câteva din proprietăţile transformărilor unitare.

- O transformare unitară conservă energia semnalului. Această proprietate o vom demonstra pentru cazul unei transformări unitare unidimensionale, pentru simplitate, deşi este perfect valabilă şi în cazul unei transformări bidimensionale. Fie $\underline{u}$ un semnal discret uni-dimensional, format din N eşantioane, şi o transformare unitară dată de matricea A. Relaţiile de transformare vor fi:

  $$\underline{v} = A \cdot \underline{u}$$
  $$\underline{u} = A^{*T} \cdot \underline{v}$$

  Energia semnalului $\underline{u}$ este dată de norma la pătrat a spaţiului în care este reprezentat semnalul:

  $$E_v = \|\underline{v}\|^2 = \underline{v}^{*T} \cdot \underline{v} = (A\underline{u})^{*T} \cdot A\underline{u} = \underline{u}^{*T} A^{*T} A\underline{u} = \underline{u}^{*T} \cdot \underline{u} = \|\underline{u}\|^2 = E_u \tag{8.22}$$

  În general transformarea se alege astfel încât energia să fie inegal distribuită în spaţiul transformatei, chiar dacă ea era uniform distribuită în spaţiul original.

- Entropia unui semnal discret cu valori aleatoare se conservă printr-o transformare unitară. Dar entropia este o măsură a cantităţii de informaţie, ceea ce înseamnă că o transformare unitară păstrează informaţia conţinută în semnal.

- Coeficienţii în spaţiul transformatei sunt decorelaţi sau aproape decorelaţi. Transformata optimă care compactează maximum de energie într-un număr dat de coeficienţi şi care în acelaşi timp decorelează complet aceşti coeficienţi, este transformarea Karhunen-Loève.

8.2 Transformata Fourier discretă

8.2.1 Transformata Fourier unidimensională

Pentru un semnal unidimensional, $\underline{u}$, de dimensiune N, de forma:

$$\underline{u} = [u(0), u(1), ..., u(N-1)]^T \tag{8.23}$$

transformarea Fourier directă este dată de relaţia:

$$v(k) = \sum_{n=0}^{N-1} u(n) \cdot e^{-\frac{2\pi jkn}{N}} \quad \vee\, k = 0..N-1 \tag{8.24}$$

iar transformarea Fourieri inversă de relaţia:

$$u(n) = \frac{1}{N} \sum_{k=0}^{N-1} v(k) \cdot e^{\frac{2\pi jkn}{N}} \quad \vee\, n = 0..N-1 \tag{8.25}$$

Astfel definită, transformarea Fourier nu este unitară. Următoarele relaţii definesc transformarea Fourier unitară, directă şi inversă:

$$v(k) = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} u(n) \cdot e^{-\frac{2\pi jkn}{N}} \quad \vee\, k = 0..N-1 \tag{8.26}$$

$$u(n) = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} v(k) \cdot e^{\frac{2\pi jkn}{N}} \quad \vee\, n = 0..N-1 \tag{8.27}$$

Dacă definim matricea $F = \{f(k,n)\}$ a transformării, având elementele:

$$f(k,n) = \frac{1}{\sqrt{N}} e^{-\frac{2\pi jkn}{N}} \quad k, n = 0..N-1 \tag{8.28}$$

atunci transformarea Fourier se poate scrie matricial astfel:

$$\underline{v} = F \cdot \underline{u} \tag{8.29}$$

$$\underline{u} = F^* \cdot \underline{v} \tag{8.30}$$

cu observaţia că matricea F are următoarea proprietate: $F = F^T$.

Pentru calculul transformatei Fourier discrete, există algoritmi rapizi (FFT[1]) care reduc complezitatea calculelor de la $O(N^2)$ la $O(NlogN)$.

8.2.2 Transformarea Fourier bidimensională

Pentru o imagine $U = \{u(m,n)\}_{m,n=0..N-1}$, de dimensiune $N \times N$, imaginea transformată $V = \{v(k,l)\}_{k,l=0..N-1}$ se calculează cu relaţia următoare, ce reprezintă transformarea Fourier în ipoteza separabilităţii:

$$v(k,l) = \frac{1}{N} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} u(m,m) \cdot e^{-\frac{2\pi j(km+ln)}{N}} \tag{8.31}$$

[1] Fast Fourier Transform

iar transformarea Fourier inversă este dată de formula:

$$u(m,n) = \frac{1}{N} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} v(k,l) \cdot e^{\frac{2\pi j(km+ln)}{N}} \tag{8.32}$$

Dacă folosim matricea F definită cu relaţia (8.28), atunci matricial se poate scrie:

$$V = F \cdot U \cdot F \tag{8.33}$$

$$U = F^* \cdot V \cdot F^* \tag{8.34}$$

8.3 Transformata cosinus discretă

Transformata cosinus este o transformată unitară separabilă, definită de matricea $C = \{c(k,n)\}$, ale cărei elemente sunt date de relaţia:

$$c(k,n) = \begin{cases} \frac{1}{\sqrt{N}}, & k = 0, 0 \leq n \leq N-1 \\ \sqrt{\frac{2}{N}} cos\frac{(2n+1)\pi k}{2N}, & 1 \leq k \leq N-1, 0 \leq n \leq N-1 \end{cases} \tag{8.35}$$

Transformata cosinus, directă şi inversă, pentru un semnal unidimensional, este dată de relaţiile:

$$v(k) = \alpha(k) \sum_{n=0}^{N-1} u(n) cos\frac{(2n+1)\pi k}{2N}, \quad 0 \leq k \leq N-1 \tag{8.36}$$

$$u(n) = \sum_{n=0}^{N-1} \alpha(k) v(k) cos\frac{(2n+1)\pi k}{2N}, \quad 0 \leq n \leq N-1 \tag{8.37}$$

unde

$$\alpha(0) = \sqrt{\frac{1}{N}}, \quad \alpha(k) = \sqrt{\frac{2}{N}}, \;\; 1 \leq k \leq N-1 \tag{8.38}$$

Transformarea cosinus bidimensională, directă şi inversă, este dată de următoarele două relaţii, scrise matricial:

$$V = C \cdot U \cdot C^T \tag{8.39}$$

$$U = C^T \cdot U \cdot C \tag{8.40}$$

deoarece matricea C are proprietatea că $C = C^*$, elementele sale fiind numere reale.

Observaţie: transformarea cosinus nu este partea reală a transformării Fourier.

8.4 Transformata sinus discretă

Transformata sinus este o transformată unitară separabilă, definită de matricea $S = \{s(k,n)\}$, ale cărei elemente sunt date de relaţia:

$$s(k,n) = \sqrt{\frac{2}{N+1}} sin\frac{(k+1)(n+1)\pi}{N+1}, \quad 0 \leq k,n \leq N-1 \tag{8.41}$$

Relaţiile ce definesc transformarea sinus unidimensională, directă şi inversă, sunt următoarele:

$$v(k) = \sqrt{\frac{2}{N+1}} \sum_{n=0}^{N-1} u(n) sin\frac{(k+1)(n+1)\pi}{N+1}, \quad 0 \leq k \leq N-1 \tag{8.42}$$

$$u(n) = \sqrt{\frac{2}{N+1}} \sum_{k=0}^{N-1} v(k) sin\frac{(k+1)(n+1)\pi}{N+1}, \quad 0 \leq n \leq N-1 \tag{8.43}$$

Transformarea sinus bidimensională, directă şi inversă, se scrie matricial astfel:

$$V = S \cdot U \cdot S \tag{8.44}$$

$$U = S \cdot V \cdot S \tag{8.45}$$

deoarece matricea S are proprietatea că $S = S^* = S^T = S^{-1}$.

Observaţie: Transformarea sinus nu este partea imaginară a transformării Fourier.

DESFĂŞURAREA LUCRĂRII

Problema 1. Pentru o imagine de dimensiune $N \times N$, adică pătrată, observaţi imaginea transformată obţinută cu ajutorul transformatei cosinus bidimensională (funcţia `transformata_cosinus_discreta` din meniul `Algoritmi`). Codul acestei funcţii este prezentat în continuare:

```
void ImageViewer :: transformata_cosinus_discreta( void )
{
   int w, h;
   int i, j, k;
   double pi = 3.1415926;

   w = image.width();
```

```
 h = image.height();

 if( w == h )
 {
    int N = w;
    double max = 0;
    double C[ N ][ N ];
//matricea transformarii cosinus

int U[ N ][ N ];
//matricea imaginii in spatiul original

double V[ N ][ N ];
//matricea imaginii in spatiul transformatei

double  AUX[ N ][ N ];

    //formarea matricei C a transformarii cosinus discreta
    for( i = 0; i < N; i++ )
       C[ 0 ][ i ] = 1. / sqrt( N );

    for( i = 1; i < N; i++ )
       for( j = 0; j < N; j++ )
       {
          C[ i ][ j ] = sqrt( 2./N ) *
               cos( pi * ( 2*j + 1 ) * i / ( 2*N ) );

          if( C[ i ][ j ] > max )
             max = C[ i ][ j ];
     }

    //formarea matricei U
    for( i = 0; i < N; i++ )
       for( j = 0; j < N; j++ )
          U[ i ][ j ] = qRed( image.pixel( i, j ));

     //V = C*U*Ct
     //mai intii vom calcula AUX = C * U
     for( i = 0; i < N; i++ )
        for( j = 0; j < N; j++ )
        {
           AUX[ i ][ j ] = 0;
           for( k = 0; k < N; k++ )
           AUX[ i ][ j ] += C[ i ][ k ] * U[ k ][ j ];
```

```
            }

        //apoi V = AUX * Ct
        max = 0;
        for( i = 0; i < N; i++ )
           for( j = 0; j < N; j++ )
           {
              V[ i ][ j ] = 0;
              for( k = 0; k < N; k++ )
                 V[ i ][ j ] += AUX[ i ][ k ] * C[ j ][ k ];

              if( V[ i ][ j ] > max )
                 max = V[ i ][ j ];
           }

        QImage transf( N, N, 32, 0, QImage::IgnoreEndian );

        for( i = 0; i < N; i++ )
           for( j = 0; j < N; j++ )
           {
              int niv = (int)( V[ i ][ j ] * 255 / max );
              transf.setPixel( i, j, qRgb( niv, niv, niv ) );
           }

     image = transf;
     pm = image;
     update();
}
```

Problema 2. Implementaţi transformarea sinus bidimensională.

Problema 3. Implementaţi transformarea Fourier bidimensională (numai partea reală, care reprezintă spectrul de frecvenţe spaţiale al imaginii).

Problema 4. Verificaţi proprietatea de conservare a energiei pentru una din transformări.