

## Lucrarea 4

# Îmbunătățirea imaginilor prin egalizarea histogramei

### BREVIAR TEORETIC

Tehnicile de îmbunătățire a imaginilor bazate pe calculul histogramei modifică histograma astfel încât aceasta să aibă o anumită formă dorită.

#### 4.1 Histograma unei imagini

Histograma unei imagini reprezintă frecvența relativă de apariție a nivelelor de gri din imagine. Pentru o imagine  $f$ , de dimensiune  $M \times N$  pixeli, histograma se definește astfel:

$$h(i) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \delta(i, f(m, n)) \quad , \quad i = 0, \dots, L - 1 \quad (4.1)$$

unde funcția  $\delta$  este definită în următorul mod:

$$\delta(x, y) = \begin{cases} 1, & \text{dacă } x = y, \\ 0, & \text{dacă } x \neq y. \end{cases} \quad (4.2)$$

Din punct de vedere statistic, putem considera valoarea fiecărui pixel al imaginii ca o realizare particulară a unei variabile aleatoare asociată nivelelor de gri, caz în care histograma reprezintă funcția de densitate de probabilitate a acestei variabile aleatoare. Fiind o funcție de densitate de probabilitate, histograma oricărei imagini verifică condiția de normare:

$$\sum_{i=0}^{L-1} h(i) = 1 \quad (4.3)$$

Practic, calculul histogramei presupune parcurgerea pixel cu pixel a imaginii și contorizarea numărului de nivele de gri întâlnite.

## 4.2 Egalizarea histogramei

Egalizarea histogramei reprezintă o operație de accentuare a contrastului și are ca scop obținerea unei histograme uniforme.

Vom asocia unui pixel din imagine o variabilă aleatoare  $\xi$ . Astfel, valorile intensității luminoase ale pixelilor reprezintă realizări particulare ale variabilei aleatoare asociate. Vom considera că variabila aleatoare  $\xi$  are o densitate de probabilitate  $w_\xi(x)$  și o funcție de repartiție  $F_\xi(x) = P\{\xi \leq x\}$ .

Vom defini în continuare variabila  $\eta$ , care are funcția de repartiție:

$$F_\eta(x) = \int_0^x p_\xi(t) dt$$

și care va fi uniform distribuită în intervalul  $(0, 1)$ .

Pentru cazul discret, presupunem că nivelele  $x$  de gri ale pixelilor au valori între 0 și  $L-1$  (unde  $L$  este de regulă 256), având asociate probabilitățile de apariție  $p_\xi(x_i)$ , unde  $x_i = 0, 1, \dots, L-1$ . Aceste probabilități pot fi estimate pe baza calculului histogramei, considerând imaginea dată, ca fiind o realizare particulară a procesului aleator descris de variabila aleatoare  $\xi$ , astfel:

$$p_\xi(x_i) = \frac{h(x_i)}{\sum_{i=0}^{L-1} h(x_i)}$$

Noile nivele de gri, reprezentând valori discrete ale variabilei  $\eta$  din intervalul  $[0, L-1]$  se vor calcula cu formulele:

$$h_c(x) = \sum_{x_i=0}^x p_\xi(x_i)$$

$$nivel_{nou} = int \left[ \frac{h_c[nivel_{vechi}] - h_{cmin}}{1 - h_{cmin}} (L-1) + 0.5 \right]$$

unde  $h_c$  reprezintă *histograma cumulativă* a imaginii, iar  $h_{cmin}$  este valoarea minimă a histogramei cumulative.

### 4.2.1 Algoritmul de egalizare a histogramei

Algoritmul de egalizare de histogramă, folosit în practică, poate fi descris în limbaj pseudocod astfel:

**Pasul 1.** Se calculează histograma imaginii:

**pentru**  $i = 1, \dots, L$

$h[i] = 0$

**pentru**  $i = 1, \dots, M$

**pentru**  $j = 1, \dots, N$

$$\begin{aligned} \text{nivel} &= \text{imagine}[i,j] \\ h[\text{nivel}] &= h[\text{nivel}] + 1 \end{aligned}$$

unde  $L$  este numărul de nivele de gri (256),  $h$  este histograma imaginii, iar  $M$  și  $N$  sunt dimensiunile imaginii.

**Pasul 2.** Se calculează histograma cumulativă a imaginii:

$$\begin{aligned} hc[1] &= h[1] \\ \text{pentru } i &= 2, \dots, L \\ hc[i] &= hc[i-1] + h[i] \end{aligned}$$

**Pasul 3.** Se calculează noile nivele de gri din imagine, sub forma unei transformări  $y = T(x)$  dată de formula:

$$y = T(x) = \left[ \frac{hc[x] - hc[1]}{NM - hc[1]} (L - 1) + 0.5 \right]$$

astfel:

$$\begin{aligned} \text{pentru } i &= 1, \dots, M \\ \text{pentru } j &= 1, \dots, N \\ \text{nivel\_vechi} &= \text{imagine}[i,j] \\ \text{nivel\_nou} &= T(\text{nivel\_vechi}) \\ \text{imagine}[i,j] &= \text{nivel\_nou} \end{aligned}$$

### 4.3 Observații

- Deși la prima vedere egalizarea de histogramă ar părea că este o operație punctuală, din cauza formulei de calcul a noilor valori de gri, ea este totuși o operație integrală, datorită faptului că pentru fiecare pixel din imagine noua valoare se calculează pe baza calculului histogramei și, deci, pe baza valorilor tuturor pixelilor din imagine.
- În Figura 4.1 puteți observa cum imaginea a fost îmbunătățită prin egalizarea histogramei.
- În Figura 4.2 puteți observa cum s-a modificat forma histogramei imaginii originale, după egalizare.



Figura 4.1: Egalizarea histogramei: (a) imaginea originală și (b) imaginea rezultată.

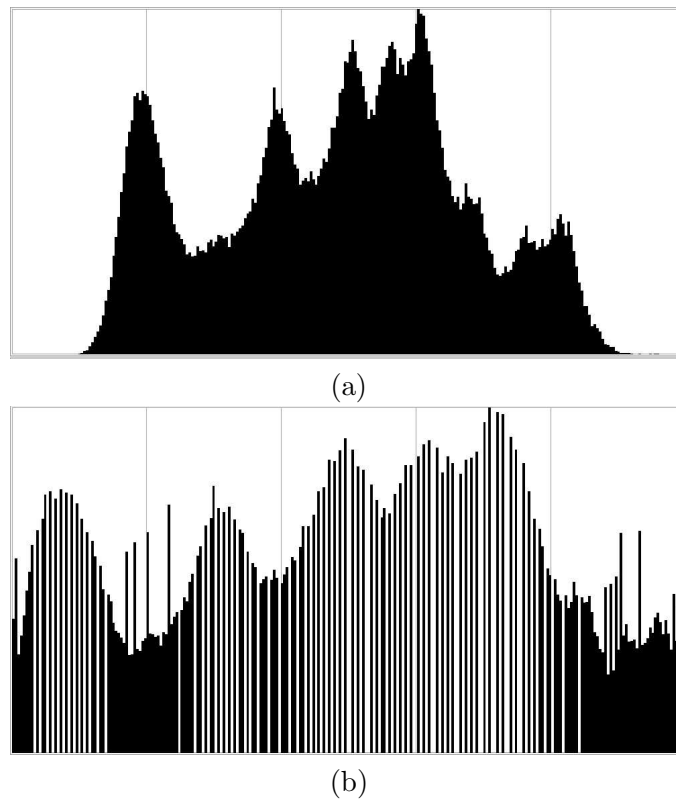


Figura 4.2: Histograma (a) originală și (b) după egalizare.

## DESFĂȘURAREA LUCRĂRII

O posibilă implementare a algoritmului de egalizare a histogramei este prezentat în continuare:

```
void ImageViewer :: egalizeaza_histograma( void )
{
    int i, j;
    int width, height;
    int h[ 256 ];

    for( i = 0; i < 256; i++ )
        h[ i ] = 0;

    width = image.width();
    height = image.height();

    //calcularea histogramei imaginii
    for( i = 0; i < width; i++ )
        for( j = 0; j < height; j++ )
        {
            QRgb pixel;
            pixel = image.pixel( i, j );

            int nivel_gri = qRed( pixel );
            h[ nivel_gri ]++ ;
        }

    //calcularea histogramei cumulative
    double hc[ 256 ];

    hc[ 0 ] = h[ 0 ];
    for( i = 1; i < 256; i++ )
        hc[ i ] = hc[ i - 1 ] + h[ i ];

    QImage imag_eq( width, height, 32, 0, QImage::IgnoreEndian );

    //egalizarea histogramei
    for( i = 0; i < width; i++ )
        for( j = 0; j < height; j++ )
        {
            QRgb pixel = image.pixel( i, j );

            int nivel = qRed( pixel );
```

```
int nivel_nou = (int)( ( hc[nivel] - hc[0] ) * 255 /
    ( width*height - hc[0] ) );

    imag_eq.setPixel( i, j,
        qRgb( nivel_nou, nivel_nou, nivel_nou ) );
}

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

**Problema 1.** Observați forma histogramei pentru câteva imagini în tonuri de gri.

**Problema 2.** Observați efectele egalizării de histogramă pentru diferite imagini, inclusiv pentru o imagine subexpusă și pentru una supraexpusă.

**Problema 3.** Modificați funcția `histograma_imagini` astfel încât aceasta să calculeze histograma cumulativă a imaginii. Observați forma unei histograme cumulative.

**Problema 4.** Justificați faptul că histograma cumulativă a unei imagini poate fi considerată estimatul unei funcții de repartiție.