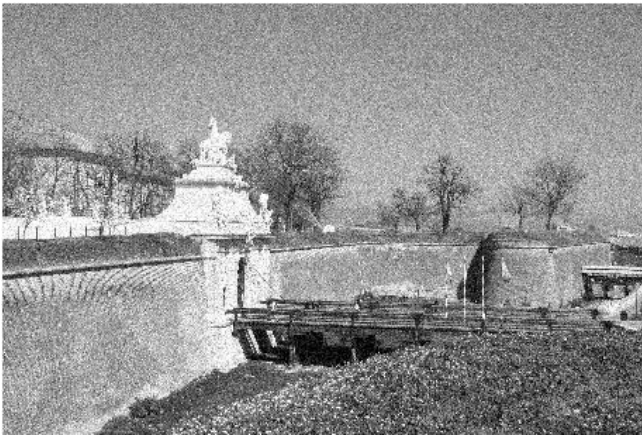


Lucrarea 5

Îndepărtarea zgomotului din imagini

Obiective: îndepărtarea zgomotului uniform folosind *filtru de mediere* (Filtru Trece Jos); îndepărtarea zgomotului de tip *sare și piper* folosind *filtru median*.

Pixelii afectați de zgomot au o intensitate diferită de cea a vecinilor lor. Această observație stă la baza multor algoritmi de îndepărtare a zgomotului.



Imagine afectată de zgomot uniform



Imagine afectată de zgomot de tip *sare și piper*

Figura 1. Imagini afectate de zgomot

Tipuri uzuale de zgomot:

- zgomot uniform = valori uniform distribuite într-un interval dat
- zgomot de tip *sare și piper* = doar valori foarte apropiate de 0 (pixeli întunecați) și valori foarte apropiate de 255 (pixeli strălucitori)

1. Filtru de mediere (FTJ)

Este util pentru filtrarea imaginilor afectate de zgomot uniform.

Filtrarea trece-jos se poate efectua prin convoluția imaginii afectate de zgomot cu o matrice:

$$H_1 = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}, \quad H_2 = \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}, \quad \dots, \quad H_n = \frac{1}{(n+2)^2} \begin{bmatrix} 1 & n & 1 \\ n & n^2 & n \\ 1 & n & 1 \end{bmatrix}.$$

Aceste matrice, numite măști de îndepărtare a zgomotului (măști de filtrare), sunt normalizate la unitate, astfel încât procesul de îndepărtare a zgomotului nu introduce o schimbare de amplitudine în imaginea rezultată.

Valoarea centrală a măștii are coordonatele $(1, 1)$. În aceste condiții, pentru o mască de dimensiune 3×3 , coordonatele elementelor măștii sunt:

$$\{(-1, -1), (-1, 0), (-1, 1), (0, -1), (0, 0), (0, 1), (1, -1), (1, 0), (1, 1)\}$$

Exemplu de realizare a filtrării cu o mască de dimensiune 3×3 .

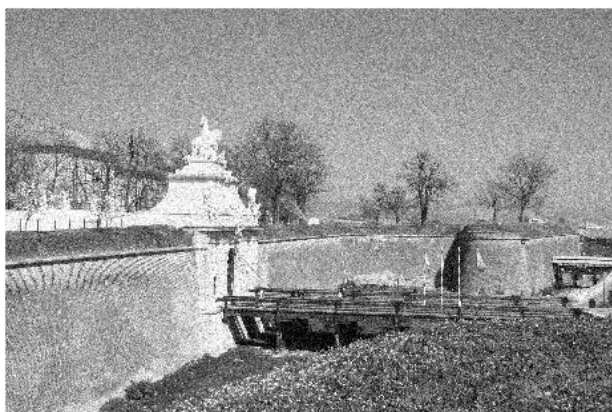
- Fie o imagine x , afectată de zgomot
- Fie imaginea y , imaginea obținută în urma filtrării
- Se parcurge pixel cu pixel imaginea ce se dorește a se filtra. Fie pixelul curent $x(i, j)$
- Se calculează pixelul curent din imaginea filtrată:

$$y(i, j) = x(i-1, j-1) \cdot (-1, -1) + x(i-1, j) \cdot (-1, 0) + x(i-1, j+1) \cdot (-1, 1) + \\ + x(i, j-1) \cdot (0, -1) + x(i, j) \cdot (0, 0) + x(i, j+1) \cdot (0, 1) + \\ + x(i+1, j-1) \cdot (1, -1) + x(i+1, j) \cdot (1, 0) + x(i+1, j+1) \cdot (1, 1)$$

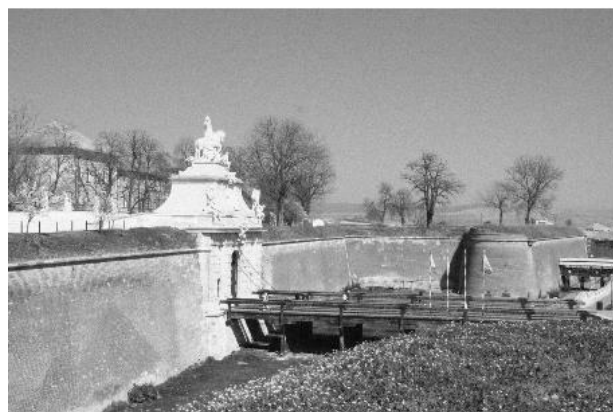
Observații:

- Dacă se folosește masca H_1 se realizează medierea imaginii
- Dacă se folosesc măștile H_2, H_3 etc se realizează o mediere ponderată a imaginii, în care pixelul central din fereastră are cea mai mare pondere.

Exemplu. Dacă se aplică medierea asupra unei imagini grayscale afectate de zgomot (ca cea din *Figura 2.a*), folosind o fereastră de mediere de 5×5 pixeli, se obține imaginea din *Figura 2.b*.



a) Imagine cu zgomot



b) Imagine filtrată

Figura 2. Exemplu de filtrare a unei imagini afectată de zgomot uniform

2. Filtru median

Este util pentru filtrarea unei imagini afectate de zgomot de tip *sare și piper*.

Exemplu de realizare a filtrării mediane cu o masca de dimensiune 3 x 3

- Fie o imagine x , afectată de zgomot de tip *sare și piper*
- Fie imaginea y , imaginea obținută în urma filtrării
- Se parcurge pixel cu pixel imaginea ce se dorește a se filtra. Fie pixelul curent $x(i, j)$
- Se pune pixelul curent $x(i, j)$ și toți cei 8 vecini ai săi într-un vector v
- Se sortează vectorul v și se înlocuiește pixelul curent $x(i, j)$ cu pixelul de pe poziția centrală din vectorul v . În cazul nostru, $x(i, j) = v(5)$.



a) Imagine cu zgomot



b) Imagine filtrată

Figura 3. Exemplu de filtrare a unei imagini afectată de zgomot de tip *sare și piper*

3. Desfășurarea lucrării

- Să se filtreze o imagine afectată de zgomot uniform folosind *filtru de mediere*
- Să se filtreze o imagine afectată de zgomot de tip *sare și piper* folosind *filtrul median*