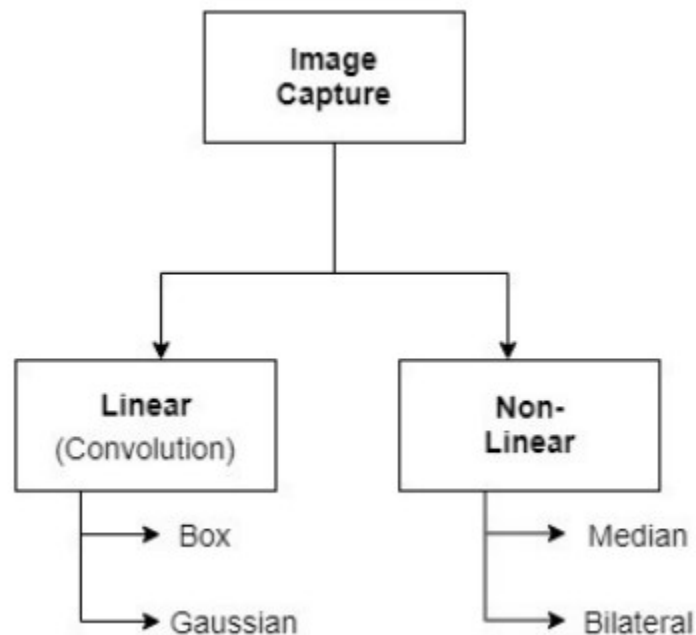


# IMAGE FILTERING - TECHNIQUES, ALGORITHM AND APPLICATIONS

Buzdugan Mihaela

Filtrarea imaginii se face pentru a îmbunătăți calitatea imaginii. De exemplu, netezirea unei imagini reduce zgomotul, imaginile blurate pot fi rectificate.

Există în general două tipuri de algoritmi, liniari și non-liniari. Filtrarea liniară se obține prin convoluție și multiplicare Fourier, în timp ce filtrarea non-liniară nu poate fi obținută prin niciuna dintre acestea. Rezultatul nu este o funcție liniară a intrării sale, astfel rezultatul variază într-un mod neintuitiv.



Filtrele liniare și non-liniare sunt algoritmi utilizați pentru filtrare. Filtru potrivit ar trebui să fie selectat pentru orice scop specific. Dacă imaginea sau intrarea dată are o cantitate mică de zgomot, dar magnitudinea este mare, atunci se folosesc filtre non-liniare, în timp ce filtrul este suficient atunci când intrarea dată conține o cantitate mare de zgomot, dar magnitudinea zgomotului este scăzută.

Filtrele liniare sunt cele mai frecvent utilizate, deoarece sunt cele mai simple și cele mai rapide. Spre deosebire de filtrele non-liniare, filtrarea liniară se face prin aplicarea algoritmului pe pixelii vecinii ai pixelilor de intrare în imagine. Pixelii din vecinătate sunt identificați prin locațiile lor care sunt relative față de pixelul de intrare.

## MEDIAN FILTER (NON LINIAR)

Filtru median este cel mai eficient algoritm de filtrare non-linear pentru a detecta și elimina zgomotul. Filtru median păstrează marginile în timp ce elimină zgomotul, de aceea este cel mai utilizat, astfel încât, în afară de a reduce zgomotul până la 60%, acesta păstrează și detaliile imaginii. Pixelii zgomotoși sunt evaluați și etichetați ca fiind pixeli zgomotoși și filtrul median bazat pe comutare se aplică la alți pixeli care nu sunt zgomotoși.



*Grayscale image*



*Median filtered image*

Implementare: Funcția "median\_filter" primește o imagine și o dimensiune a kernel-ului ca parametri. Ea creează o copie a imaginii și o numește "filtered\_image". Apoi, parcurge fiecare pixel al imaginii, ia pixelii din jurul său, îi sortează și atribuie pixelul din mijloc ca fiind valoarea pixelului curent din "filtered\_image".

## BILATERAL FILTER (NON LINIAR)

Filtru bilateral este un tip de filtru non-linear, care reduce zgomotul prin netezire în timp ce păstrează marginile. El ia suma ponderată a pixelilor care sunt apropiați de fiecare pixel și înlocuiește intensitatea pixelilor cu media acelei sume ponderate. În această metodă, nu se produc culori fantomă în lungul marginilor imaginilor de ieșire. Filtrarea bilaterală este similară cu convoluția Gaussiană, deoarece este media pixelilor, dar filtrul bilateral ia în considerare variațiile de intensități, astfel păstrând marginile imaginii.



*Grayscale image*



*Bilateral filtered image*

**Implementare:** Funcția "bilateral\_filter" creează o imagine goală pentru a stoca rezultatul filtrării și calculează valoarea fiecărui pixel prin aplicarea unui filtru bilateral în jurul fiecărui pixel. Filtrarea bilaterală se bazează pe aplicarea unui filtru Gaussian pe distanța geometrică și pe distanța de intensitate a pixelilor din jur. În final, returnează imaginea filtrată.

### GAUSSIAN FILTER (LINIAR)

Un filtru Gaussian este un tip de filtru care poate fi utilizat pentru a genera un nucleu. Un nucleu este o matrice mica care este o matrice convoluționată utilizată în diverse tehnici de filtrare a imaginilor. Kernel-ul este generat bazat pe ecuația unei funcții gaussiene. Gaussian blur poate fi aplicat pentru operații de netezire sau filtrare. Imaginea filtrată are mai multe detalii comparativ cu un filtru de mediu de blur. Acesta oferă rezultate mai bune decât box blur. Într-un nucleu particular, greutatea atribuită în estomparea Gaussiană se bazează pe distanța pixelilor din apropiere de pixelul curent, în timp ce bilateral folosește intensitatea pixelilor pentru filtrarea imaginii.



*Grayscale image*



*Gaussian filtered image*

Implementare: Funcția "gaussian\_filter" primește o imagine și o valoare sigma ca parametri. Ea creează un kernel utilizând funcția Gaussiană 2D, apoi aplică acest kernel prin convoluție pe imagine pentru a o filtra. Rezultatul este o imagine scalată și returnată ca fiind imaginea filtrată.

### BOX BLUR FILTER (LINIAR)

În box blur, o imagine cu valori de pixel 9x9 poate fi considerată având valori de vecinătate 3x3. Netezirea se realizează prin calcularea mediei valorilor pixelilor din vecinătate a pixelului particular în imaginea de ieșire. În acest fel, pixelii cu intensitate mai mare se convertesc într-un pixel cu valoare mai mică și invers, astfel echilibrând pixelii imaginii.



*Grayscale image*



*Box blur filtered image*

Implementare: Funcția "box\_blur" primește o imagine și o dimensiune a kernel-ului ca parametri. Ea creează o copie a imaginii și o numește "filtered\_image". Apoi, parcurge fiecare pixel al imaginii și aplică un filtru de medie aritmetică pe fiecare pixel folosind pixelii din jurul său. În final, returnează "filtered\_image" ca fiind imaginea filtrată. În acest exemplu, dimensiunea kernel-ului este setată la 4 și este aplicat pe imaginea "gray\_image" pentru a obține "box\_blur\_image".

## ANALIZA

Pentru analiza am folosit MSE(mean-square error) si PSNR(peak signal-to-noise ratio).

```
MSE for Blur: 560.9344224329936
MSE for Box blur: 4870.089830755418
MSE for Gaussian : 527.5376588473422
MSE for Median: 50.72139810752867
MSE for Bilateral: 43.5883616493225
PSNR for Blur: 20.641682690238472
PSNR for Box blur: 11.255433888439175
PSNR for Gaussian: 20.908268932094202
PSNR for Median: 20.908268932094202
PSNR for Bilateral: 31.737098153196122
```

Valorile MSE mai mici indică o performanță mai bună în ceea ce privește păstrarea integrității imaginii originale, în timp ce valorile PSNR mai mari indică o performanță mai bună în ceea ce privește reducerea zgomotului.

Din analiza făcută, putem observa că filtrele non-liniare au o performanță mai mare atât în păstrarea imaginii originale, cât și în ceea ce privește reducerea zgomotului. Având în vedere beneficiile prezentate, trebuie ținut cont că un filtru liniar este mai rapid și mai simplu.

Box Blur



Gaussian



Median



Bilateral Image

