

# STUDIUL TEHNICILOR DE RECUNOASTERE SI CLASIFICARE A IMAGINILOR 2D

Barbu Vlad Alexandru

Facultatea de Matematica si Informatica,  
Universitatea din Bucuresti,  
Bucuresti, Romania  
vlad.barbu@s.unibuc.ro

**ABSTRACT.** Automatizarea clasificarii imaginilor si recunoasterea predictibila a formelor de interes din contextul acestora reprezinta o problema complexa, avand implicatii majore in multe domenii stiintifice. Definirea unui proces sistematic de prelucrare, identificare si clasificare reprezinta un demers necesar abordarii acestei probleme. Acest raport incearca sa ofere o viziune de ansamblu asupra solutiilor existente, avand drept obiect de studiu pentru analiza si implementarea acestora - recunoasterea si extragerea informatiei din cadrul unui document scos la imprimanta (deseori pornind de la structura unui tipizat de baza, fiind completat ulterior in format digital sau cu scris de mana).

**Cuvinte cheie:** Recunoasterea formelor. Prelucrarea imaginilor. Clasificarea imaginilor 2D.

## 1 Introducere

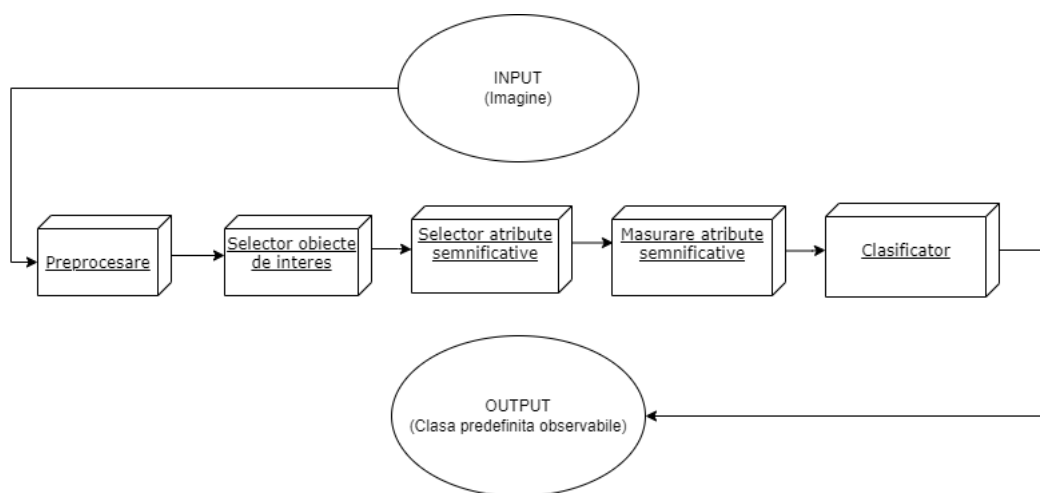
**Procesul de recunoastere a formelor** poate fi vazut drept o inlantuire de proceduri cu ajutorul carora se poate ajunge la un rezultat concludent de clasificare a unui obiect de interes din cadrul unei multimi de entitati observabile. O astfel de metoda de recunoastere devine utila mai ales atunci cand abordarea directa este imposibila - atunci cand o clasificare manuala nu este fezabila (in cazul in care nevoia de recunoastere este una recurenta).

**Descrierea unui proces robust de clasificare automata a imaginilor** consta in stabilirea unui set de reguli in baza caruia, datele de intrare pot fi grupate in clase identificabile. O clasa identificabila este reprezentata de o multime finita de factori cantitativi si calitativi cu ajutorul carora unui obiect observabil ii este asociat un numar de attribute semnificative - contextul de identificare a unei astfel de clase este determinat de o colectie predefinita de metadate. Metoda folosita pentru delimitarea claselor identificabile reprezinta un caz particular specific domeniului de interes al aplicatiei concrete.

## 2 Descrierea unui sistem de recunoastere

Un sistem robust si eficient de recunoastere a formelor va oferi rezultate corecte, predictibile urmand o inlantuire de proceduri de transformare, prelucrare, extractie si clasificare. Componentele principale ale unui sistem de recunoastere si clasificare a formelor sunt urmatoarele:

- **Componenta de preprocesare** - cu ajutorul acestei componente se vor normaliza datele de intrare (mediul de provenienta al acestora fiind necunoscut, stabilirea unui standard este necesara) si se vor evidentia obiectele de interes prin eliminarea impuritatilor
- **Selectorul de obiecte de interes** - in cadrul acestei etape, imaginea va fi parcursa si redusa la o multime de entitati de interes (structura observabilelor va fi definita prin intermediul acestei componente, urmand ca aceasta sa fie definitia standardului de metadata folosit de-a lungul procesului de recunoastere)
- **Selectorul de attribute semnificative** - cu ajutorul acestuia, vom putea identifica factorii cantitativi si calitativi pe care un anumit observabil ii respecta (acesti factori trebuie sa fie predefiniti cu ajutorul unor descriptori relevanti domeniului de interes al aplicatiei)
- **Componenta de masurare a atributelor** - responsabilitatea acestei componente este aceea de a calcula "importanta" atributelor asociate fiecarui observabil prin invocarea unui set predefinit de metrice si metode discriminative relevante pentru fiecare context de identificare, urmata de agregarea rezultatelor obtinute
- **Clasificatorul** - etapa finala a procesului de recunoastere este reprezentata de efectuarea unei alegeri prin compararea masuratorilor efectuate in pasii anteriori pentru fiecare clasa identificabila predefinita



### 3 Reprezentarea imaginilor

Din punct de vedere matematic, o imagine poate fi reprezentata sub forma unei functii de doua variabile:

- In cazul imaginilor formate doar din tonuri de gri (i.e. Grayscale), valorile functiei reprezinta luminanta pixelilor

$$f(x, y) : R^2 \rightarrow R \quad (1)$$

- In cazul imaginilor color, valorile reprezinta un vector de 3 elemente, componente ale spatiului de culori ales (un exemplu ar fi spatiul de culori RGB: unde R - red, G - green, B - blue)

$$f(x, y) : R^2 \rightarrow R^3 \quad (2)$$

Inainte de a putea prelucra o imagine, aceasta va trebui discretizata din punct de vedere spatial. Procesul aferent discretizarii spatiale a coordonatelor (i.e. *esantionare*), descrie un mod de aproximare a unei imagini continue de tipul  $f(x,y)$ , cu o matrice 2-dimensională de tipul  $M \times N$ , unde  $M$  reprezinta numarul de randuri, iar  $N$  reprezinta numarul de coloane de pixeli.

$$f(x, y) = p_{x,y} \quad (3)$$

$$\begin{bmatrix} p_{0,0} & p_{0,1} & p_{0,2} & \dots & p_{0,m-1} \\ p_{1,0} & p_{1,1} & p_{1,2} & \dots & p_{1,m-1} \\ \dots & \dots & \dots & \dots & \dots \\ p_{n-1,0} & p_{n-1,1} & p_{n-1,2} & \dots & p_{n-1,m-1} \end{bmatrix}$$

Trebuie sa tinem cont de faptul ca discretizarea unei imagini este un proces insotit de zgomot - *eroare de cuantizare*. Una dintre cele mai folosite metode de cuantizare este cuantizarea uniforma - intervalele functiei de cuantizare sunt egale.

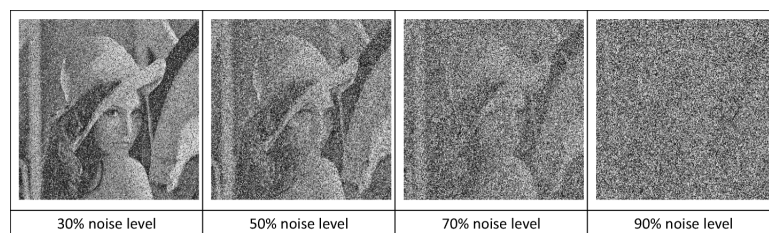
**O imagine digitala** este reprezentata de o structura de date bidimensionala, unde un element  $(x, y)$  al acesteia poarta numele de pixel.

## 4 Preprocesarea imaginii si eliminarea zgomotului

Zgomotul de imagine este o variație aleatorie a luminozității sau a informațiilor de culoare din imagini și este, de obicei, un aspect al zgomotului electronic. Acesta reprezintă un produs secundar nedorit al captării imaginii, care ascunde informațiile dorite.

Sensul original al cuvântului "zgomot" era "semnal nedorit". Prin analogie, fluctuațiile electrice nedorite sunt, de asemenea, numite "zgomot".

Zgomotul de imagine poate varia de la pete aproape imperceptibile pe o fotografie digitală făcută în condiții de lumină bună, până la imagini care sunt aproape în întregime zgomot, din care se poate obține o cantitate mică de informații chiar și prin metode de procesare sofisticată. Un astfel de nivel de zgomot ar fi inacceptabil într-o fotografie, deoarece ar fi imposibil să se determine subiectul.



Procese de eliminare a zgomotului din cadrul unei imagini sunt cunoscute și sub numele de operații de filtrare. Scopul acestora este reprezentat de nevoia evidentierii muchiilor și reliefarea entităților observabile.

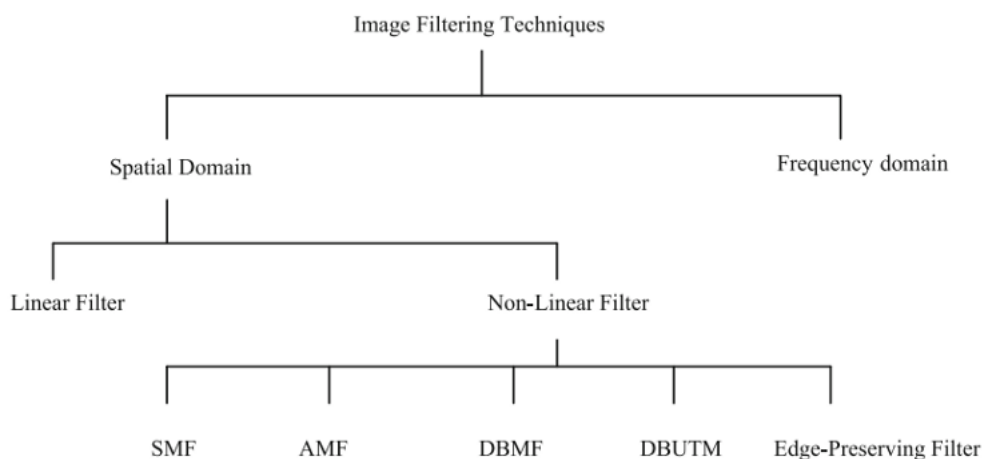
Trei dintre cele mai folosite metode de filtrare sunt:

- Filtrul trece-jos - procedura de filtrare a zgomotului realizat prin operații de uniformizare a spectrului imaginii
- Filtrul trece-banda - procedura comună de eliminare a zgomotului în domeniile în care se prelucrează imagini obținute prin metode de teledetectie
- Filtrul trece-sus - procedura utilizată în cazul în care evidențierea conturilor este principala nevoie de prelucrare

De asemenea, metodele de filtrare și analiza a semnalelor unidimensionale pot fi extinse cu ușurință în mediul semnalelor bidimensionale - putem descompune un semnal bidimensional în sume de semnale sinusoidale.

Tehnicile de filtrare sunt utilizate pentru a îmbunătăți și modifica imaginile digitale. De asemenea, filtrele de imagine sunt utilizate pentru a reduce neclaritatea și zgomotul, pentru a îmbunătăți claritatea și pentru a detecta marginile.

Filtrele de imagine sunt utilizate în principiu pentru suprimarea frecvențelor înalte (tehnici de netezire) și joase (îmbunătățirea imaginii, detectarea marginilor).



În conformitate cu această clasificare, filtrele de imagine pot fi împărțite în două categorii principale:

- Filtrarea spațială - metoda tradițională de filtrare a imaginilor, care se utilizează direct pe pixelii imaginii
- Filtrele în domeniul frecvenței - metoda utilizată pentru a elimina frecvențele înalte și joase și pentru netezire

Filtrele neliniare sunt utilizate pentru a detecta marginile. Aceste tehnici de filtrare sunt mai eficiente decât filtrele liniare. În cazul filtrării liniare, detaliile și marginile imaginii tind să se estompeze.

Filtrul Gaussian, filtrul Laplacian și filtrul de medie de vecinătate (medie) pot fi identificate ca exemple de filtre liniare. Filtrele mediane sunt filtre neliniare.

## 4.1 Filtrul median

Filtrul median este un filtru neliniar care înlocuiește valorile fiecărui pixel cu valorile mediane ale pixelilor săi vecini.

Acesta este un mod eficient de a elimina zgomotul de tip "salt and pepper" - o formă de zgomot, cunoscută și sub numele de zgomot de impuls, poate fi cauzat de perturbații bruște în semnalul de imagine și se prezintă sub forma unor pixeli albi și negri cu apariție dispersată.

30	35	40	42	42
35	42	37	37	40
38	39	40	41	42
40	41	42	43	43
42	43	45	44	46

		41		

37, 37, 39, 40, **41**, 42, 42, 43



## 4.2 Filtrul Laplacian

Tehnica de netezire Laplace este utilizată în principal pentru a detecta marginile imaginii. Aceasta evidențiază discontinuitățile nivelului de gri. Se bazează pe a doua derivare spațială a unei imagini.

Pentru a defini operatorul Laplacian, a fost utilizată ecuația de mai jos:

$$\text{Laplace}(f) = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

Detectorul de margini Laplace utilizează un singur nucleu. Pentru a detecta marginile unei imagini, acest nucleu detectează derivatele de ordinul 2 ale nivelurilor de intensitate ale imaginii, utilizând o singură trecere.

Putem utiliza cel de-al doilea nucleu din imaginile de mai jos pentru a detecta marginile cu diagonale. Acesta va oferi o aproximare mai bună. De asemenea, metoda Laplace oferă calcule mai rapide decât celelalte.

0	-1	0
-1	4	-1
0	-1	0

Kernel 1

-1	-1	-1
-1	8	-1
-1	-1	-1

Kernel 2



Kernel 1



Kernel 2

### 4.3 Filtrul Gaussian

Acest filtru este un operator convoluțional 2-D. Se utilizează pentru a estompa imaginile. De asemenea, elimină detaliile și zgomotele. Filtrul gaussian este similar cu filtrul mediu, dar principala diferență este că filtrul gaussian utilizează un nucleu. Acest nucleu are forma unei "cocoșe" gaussiene.

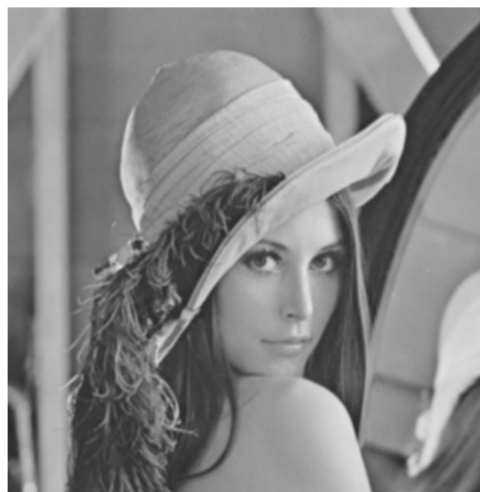
Nucleul gaussian ponderează pixelii din centrul său mult mai puternic decât cei de la limitele sale. Există diferite nuclee gaussiene. În funcție de dimensiunea nucleului, imaginea de ieșire va fi diferită.

1/16	1	2	1
	2	4	2
	1	2	1
3x3 kernel			

1/173	1	4	7	4	1
	4	16	26	16	4
	7	26	41	26	7
	4	16	26	16	4
	1	4	7	4	1
5x5 kernel					



3X3 Kernel



5x5 Kernel



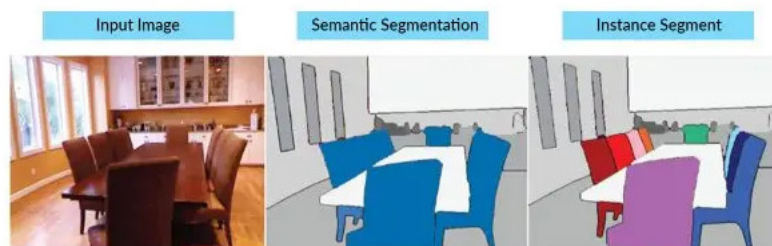
## 5 Segmentarea imaginii

Segmentarea imaginilor este procesul prin care o imagine digitală este împărțită în diferite subgrupuri (de pixeli) numite entități observabile, reducând complexitatea imaginii, astfel analiza acesteia devenind mai simplă.

Procesele de acest tip utilizează diverși algoritmi de segmentare a imaginilor pentru a diviza și grupa un anumit set de pixeli din imagine. În acest fel, atribuim de fapt "etichete" pixelilor, iar pixelii cu aceeași etichetă formează un obiect de interes.

Cu ajutorul acestor etichete, putem preciza limitele și conturul, separând entitățile observabile dintr-o imagine de "fundalul" acesteia. În exemplul de mai jos, dintr-o imagine inițială, se încearcă obținerea componentelor majore (de exemplu, scaunul, masa etc.) folosind diverse tehnici de segmentare.

Procesul de segmentare are ca scop delimitarea zonelor de interes din cadrul unei imagini, pornind de la o mulțime de criterii prestabilite. La finalul acestei operații fiecărui pixel îi va fi asociată o valoare cu ajutorul căreia îi poate fi definită apartenența la o anumită regiune de interes din imagine - o astfel de zonă de interes reprezintă un segment, iar imaginea inițială este reprezentată de reuniunea tuturor segmentelor rezultate.



De obicei, metodele de segmentare au în vedere următoarele cazuri:

- discontinuitatea entităților observabile - în acest caz o metodă potrivită de segmentare ar fi detectia conturului
- similitudinea observabilelor - cele mai comune metode de segmentare folosite în acest caz sunt metoda regiunilor și metoda pragului

## 6 Detectarea contururilor

Un contur poate fi reprezentat de o modificare brusca de lumina, culoare, umbra sau textura din contextul unei imagini digitale. Analiza imaginilor folosind metode de detectare a contururilor ajuta atat la solutionarea necesitatii de a extragere anumite obiecte de interes din cadrul acestora cat si la filtrarea informatiei inutile.

Deoarece calitatea rezultatelor oferite de procesele de detectare poate fi afectata de cantitatea de zgomot prezenta in imagine, un procedeu robust si bine stabilit de preprocesare si filtrare a acesteia este deseori necesar.

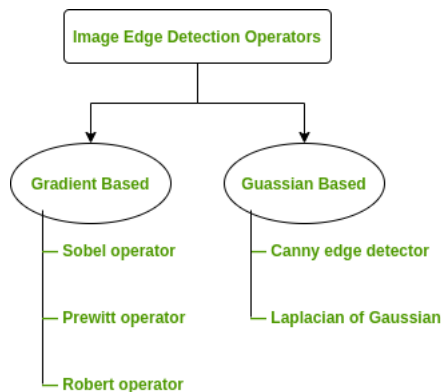
Muchiile sunt modificări locale semnificative de intensitate într-o imagine digitală. O muchie poate fi definită ca un set de pixeli conectați care formează o limită între două regiuni disjuncte. Există trei tipuri de muchii: orizontale, verticale si diagonale.

Detectarea marginilor este o metodă de segmentare a unei imagini în regiuni de discontinuitate. Este o tehnică utilizată pe scară largă în prelucrarea digitală a imaginilor, cum ar fi recunoasterea formelor ("pattern recognition") sau extragerea caracteristicilor ("feature extraction").

Detectarea marginilor permite utilizatorilor să observe caracteristicile unei imagini pentru o schimbare semnificativă a nivelului de gri. Această textură indică sfârșitul unei regiuni din imagine și începutul alteia. Aceasta reduce cantitatea de date dintr-o imagine și păstrează proprietățile structurale ale imaginii.

Operatorii de detectare a marginilor sunt de două tipuri:

- Gradient - operator bazat pe care calculează derivatele de ordinul întâi într-o imagine digitală, cum ar fi: operatorul Sobel, operatorul Prewitt, operatorul Robert
- Gaussian - operator bazat pe Gaussian care calculează derivatele de ordinul doi într-o imagine digitală, cum ar fi: detectorul de margini Canny, Laplacianul lui Gaussian



## 6.1 Operatorul Sobel

Operatorul Sobel este un operator de diferențiere discretă. Acesta calculează aproximarea gradientului funcției de intensitate a imaginii pentru detectarea marginilor imaginii.

Aplicat pe pixelii unei imagini, operatorul Sobel produce fie normala la un vector, fie vectorul gradient corespunzător. Acesta Utilizează două nuclee sau măști 3 x 3 care sunt convolute cu imaginea de intrare pentru a calcula aproximările derivatei verticale și, respectiv, orizontale.

-1	0	1
-2	0	2
-1	0	1

**Vertical**

1	2	1
0	0	0
-1	-2	-1

**Horizontal**

Avantaje:

- Calcul simplu și eficient în timp
- Foarte ușor la căutarea de margini netede

Limitari:

- Punctele de direcție diagonală nu sunt păstrate întotdeauna
- Foarte sensibil la zgomot
- Nu este foarte precis în detectarea marginilor
- Detectarea cu margini groase și aspre nu dă rezultate corespunzătoare

## 6.2 Operatorul Prewitt

Operatorul Prewitt este similar cu operatorul sobel. De asemenea, acesta detectează marginile verticale și orizontale ale unei imagini. Este una dintre cele mai bune modalități de a detecta orientarea și magnitudinea unei imagini.

<b>-1</b>	<b>0</b>	<b>+1</b>
<b>-1</b>	<b>0</b>	<b>+1</b>
<b>-1</b>	<b>0</b>	<b>+1</b>

**G<sub>x</sub>**

<b>+1</b>	<b>+1</b>	<b>+1</b>
<b>0</b>	<b>0</b>	<b>0</b>
<b>-1</b>	<b>-1</b>	<b>-1</b>

**G<sub>y</sub>**

Avantaje:

- Performanță bună la detectarea marginilor verticale și orizontale
- Cel mai bun operator pentru a detecta orientarea unei imagini

Limitari:

- Mărimea coeficientului este fixă și nu poate fi modificată
- Punctele de direcție diagonală nu sunt păstrate întotdeauna

### 6.3 Operatorul Robert

Operatorul Robert calculează suma pătratelor diferențelor dintre pixelii adiacenți pe diagonală dintr-o imagine prin diferențiere discretă, după care va face aproximarea gradientului.

<b>+1</b>	<b>0</b>
<b>0</b>	<b>-1</b>

**G<sub>x</sub>**

<b>0</b>	<b>+1</b>
<b>-1</b>	<b>0</b>

**G<sub>y</sub>**

Avantaje:

- Detectarea marginilor și orientarea sunt foarte ușoare
- Punctele de direcție diagonală sunt păstrate

Limitari:

- Foarte sensibil la zgomot
- Nu este foarte precis în detectarea marginilor

## 6.4 Operatorul Marr-Hildreth

Operatorul Marr-Hildreth sau Laplacianul lui Gauss (LoG) este un operator bazat pe gaussian care utilizează Laplacianul pentru a lua a doua derivată a unei imagini. Acesta funcționează foarte bine atunci când tranziția nivelului de gri pare a fi bruscă.

Funcționează pe baza metodei de trecere prin zero, adică atunci când derivata de ordinul doi trece prin zero, atunci acea locație specifică corespunde unui nivel maxim. Aceasta se numește locație de margine. Aici, operatorul gaussian reduce zgomotul, iar operatorul laplacian detectează marginile ascuțite.

### Laplacian of Gaussian (Marr-Hildreth Operator)

$$\text{Gaussian: } g(x, y) = \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)$$

Let:

$$h(x, y) = \nabla^2 g(x, y) = \frac{x^2 + y^2 + 2\sigma^2}{\sigma^4} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)$$

$$\begin{aligned} \text{Then: } \nabla^2 f(x, y) &= \nabla^2 g(x, y) * f(x, y) \\ &= h(x, y) * f(x, y) \end{aligned}$$

Avantaje:

- Ușor de detectat marginile și diferitele orientări ale acestora
- Există caracteristici fixe în toate direcțiile

Limitari:

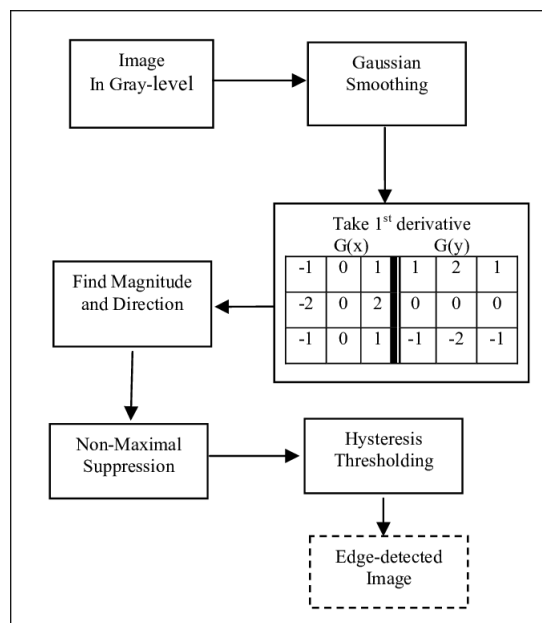
- Foarte sensibil la zgomot
- Eroarea de localizare poate fi severă la marginile curbate
- Acesta generează răspunsuri zgomotoase care nu corespund unor muchii, așa-numitele "muchii false"

## 6.5 Operatorul Canny

Operatorul Canny este un operator bazat pe gaussian pentru detectarea marginilor. Acest operator nu este sensibil la zgomot. Acesta extrage caracteristicile imaginii fără a afecta sau modifica caracteristicile.

Detectorul de margini Canny are un algoritm avansat derivat din munca anterioară a operatorului Marr-Hildreth. Este o tehnică de detectare optimă a marginilor, utilizată pe scară largă. Acesta detectează marginile pe baza a trei criterii:

- Rata de eroare scăzută
- Punctele de margine trebuie să fie localizate cu precizie
- Ar trebui să existe doar un singur răspuns pe muchie



Avantaje:

- Are o localizare bună
- Acesta extrage caracteristicile imaginii fără a modifica caracteristicile
- Mai puțin sensibil la zgomot

Limitari:

- Există o trecere falsă la zero
- Calcul complex și consumator de timp

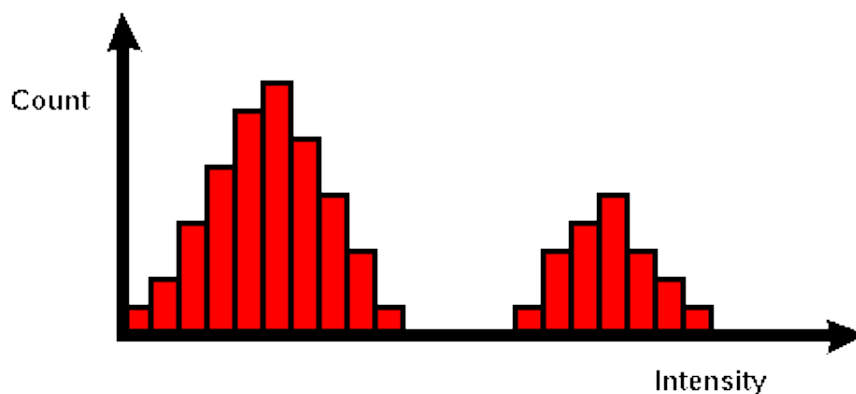
## 7 Histograme de intensitate

### 7.1 Scurta descriere

În contextul procesării imaginilor, histograma este folosită pentru reprezentarea distribuției valorilor intensității pixelilor. Această reprezentare este un grafic care expune numărul de pixeli dintr-o imagine aparținând unei valori diferite de intensitate găsite în acea imagine.

Pentru o imagine în tonuri de gri pe 8 biți, există 256 de intensități diferite posibile, astfel încât histograma va afișa grafic 256 de numere care descriu distribuția pixelilor între aceste valori ale tonurilor de gri. Histogramele pot fi, de asemenea, realizate pentru imaginile color - fie se pot realiza histogramme individuale fiecărui canal (în cazul spectrului RGB - roșu, verde și albastru), fie se poate realiza o histogramă tridimensională, în care cele trei axe reprezintă canalele roșu, albastru și verde, iar luminozitatea din fiecare punct reprezintă numărul de pixeli.

Rezultatul exact al operațiunii depinde de implementare. Acesta poate fi pur și simplu o imagine a histogramei solicitate într-un format de imagine adecvat sau poate fi un fișier de date care să reprezinte informația încapsulată de histograma.



### 7.2 Cum functioneaza

Procesul de aplicare a acestei metode este foarte simplu: imaginea este scanată într-o singură trecere (rand cu rand) și se va stoca o contorizare a numărului de pixeli găsiți pentru fiecare valoare de intensitate. Aceasta contorizare este apoi utilizată pentru a construi o histogramă reprezentativă.



## 8 Exemplu de aplicatie

Exemplul de aplicatie folosit pentru a demonstra cateva dintre tehnicile de preprocesare, recunoastere si clasificare descrise anterior este reprezentat de un microserviciu scris in limbajul 'Go' al carui scop este extragerea textului digital din cadrul unui document scanat.

De asemenea a fost realizata si o interfata pentru studiul si analiza eficientei microserviciului reprezentata de o aplicatie web care pune la dispozitie o panza pe care putem desena diverse forme sau incarca imagini de pe disc si invoca functionalitati puse la dispozitie de microserviciul de recunoastere.

Pentru mai multe detalii legate de implementari, acestea sunt puse la dispozitie pentru eventualii doritori ( [github.com/vlad-a-barbu/gocr](https://github.com/vlad-a-barbu/gocr) si [github.com/vlad-a-barbu/gocr-ui](https://github.com/vlad-a-barbu/gocr-ui)).

Structura si modul de functionare al microserviciului este descris de urmatoorii pasi:

- Preprocesarea documentului
- Extragerea obiectelor de interes (subimagini reprezentand caracterele ce urmeaza a fi identificate)
- Pentru fiecare candidat al procesului de recunoastere se vor genera cate doua histograme reprezentand distributia pixelilor de interes atat din perspectiva verticala cat si din perspectiva orizontala
- Pentru fiecare candidat al procesului de recunoastere se va genera o pereche de expresii regulate reprezentand numarul de intervale continue de pixeli de interes continuti pe randurile deprinse din perspectiva verticala, respectiva din perspectiva orizontala
- Microserviciul se va folosi de un set de histograme si perechi de expresii de referinta in baza carora se va face clasificarea

\* Setul de date de referinta este rezultat din incarcarea unei imagini de baza care va contine cate o subimagine reprezentativa fiecarui caracter folosit in documentele ce urmeaza sa fie analizate - spre exemplu imaginea de baza ar putea fi reprezentata de alfabetul limbajului in care au fost scrise documentele ce urmeaza a fi identificate. Pentru simplitate se pot folosi reprezentari ale alfabetului englez in diverse fonturi des intalnite: helvetica, arial, times new roman, etc.

\* Scopul microserviciului nu este acela de a oferi consumatorilor sai o metoda robusta si generala de identificare si clasificare. Acesta serveste drept aplicatie demonstrativa, suport de invatare si experimentare pentru diverse metode de procesare a imaginilor digitale.

## 8.1 Metode de preprocesare

Imagine folosita de-a lungul procesului de recunoastere:

Județul ..... 201..... luna ..... ziua .....

Localitatea .....

Unitatea sanitară .....

CNP 

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

**BILET DE TRIMITERE**

către: .....

Numele ..... Prenumele .....

Sexul  $\frac{M}{F}$ , în vârstă de ..... ani, cu domiciliul în :

Județul ..... Localitatea .....

Str. .... nr. ....

Diagnostic prezumtiv .....

Motivul trimeriei.....

.....

Investigații și tratamente .....

.....

.....

.....

Semnătura și parafa medicului,

Putem observa efectele conversiei la imagine tip Grayscale - a fost realizata maparea din spectrul de culori RGB in spectrul Grayscale (tonuri de gri) folosind urmatoarea formula de calcul a luminantei relative:

$$Y = 0.2126 * R_{lin} + 0.7152 * G_{lin} + 0.0722 * B_{lin}$$

Județul

Efectele aplicarii unui filtru tip threshold alegand o valoare maxima acceptata pentru luminanta pixelilor (se observa fenomenul de fragmentare a caracterelor):



Efectele aplicarii unei masti de interpolare pe verticala si orizontala a pixelilor vecini pentru a rezolva problema fragmentarii:



Efectele aplicarii unei masti de interpolare pe verticala, orizontala si diagonala a pixelilor vecini pentru a rezolva problema fragmentarii (metoda mai agresiva, se poate observa fenomenul de "lipire" a entitatilor observabile):

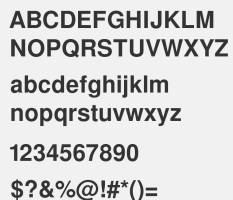


Selectarea obiectelor de interes a fost realizata prin procesul de parcurgere a imaginii preprocesate rand cu rand, si aplicarea metodelor de segmentare descrise anterior pentru a grupa pixelii de interes in mai multe clase de entitati observabile.

Pentru implementare s-au folosit algoritmi de tip 'Fill' la momentul interceptarii unui pixel de interes, luand in considerare atat vecinii de pe orizontala si verticala cat si vecinii de pe diagonala (proces eficientizat de mastile si filtrele descrise mai sus).

## 8.2 Formarea descriptorilor de referinta

Urmatoarea imagine prezinta alfabetul englez de caractere reprezentata de fontul Helvetica (unul dintre cele mai populare fonturi pentru documentele digitale):



ABCDEFGHIJKLM  
NOPQRSTUVWXYZ  
abcdefghijklm  
nopqrstuvwxyz  
1234567890  
\$?&%@!#\*()=

In baza acestei imagini s-a extras pe rand fiecare caracter, si i-au fost asociate urmatoarele metadate:

- O pereche de expresii regulate reprezentand frecventele pixelilor de interes pe randuri si pe coloane
- O histograma urmarind distributia pixelilor de interes din perspectiva orizontala
- O histograma urmarind distributia pixelilor de interes din perspectiva verticala

Histogramele vor fi folosite drept factori cantitativi in procesul de clasificare, urmand ca expresiile sa intervina drept factor calitativ (descriptori de 'forma' pentru caracterele analizate).

Exemple de expresii regulate de referinta:

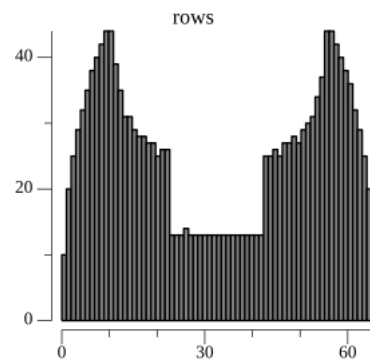
```
var imageDescriptors = map[ImageDescriptor]rune{
  { RowExpr: "1+2+1+2+", ColExpr: "1*2+[2-3]{2,}1+": 'a',
    { RowExpr: "1{3,}2{2,}1*", ColExpr: "1+2{2,}1+": 'b',
      { RowExpr: "1+2*1{3,}2*1+", ColExpr: "1+2{3,}": 'c',
        { RowExpr: "1{3,}2{6,}", ColExpr: "1+2+1+": 'd',
          { RowExpr: "1+2+1{3,}2*1", ColExpr: "13{2,}2*": 'e',
            { RowExpr: "2{4,}1+", ColExpr: "1{4,}": 'u',
              { RowExpr: "1{6,}", ColExpr: "1{2,}": 'l',
                • { RowExpr: "1{6,}2+1*", ColExpr: "1{3,}": 'j',
```

### 8.3 Generarea histogramelor

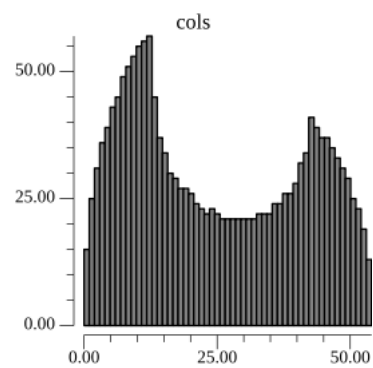
Subimagine a caracterului de referinta analizat:



Histograma urmarind distributia pixelilor de interes din perspectiva orizontala (frecventa acestora pe randuri):



Histograma urmarind distributia pixelilor de interes din perspectiva verticala (frecventa acestora pe coloane):



## References

- [1] [en.wikipedia.org/wiki/ImageNoise](https://en.wikipedia.org/wiki/ImageNoise)
- [2] [analytixlabs.co.in/blog/what-is-image-segmentation/](https://analytixlabs.co.in/blog/what-is-image-segmentation/)
- [3] [medium.com/@shashikadilhani97/digital-image-processing-filters](https://medium.com/@shashikadilhani97/digital-image-processing-filters)
- [4] [geeksforgeeks.org/image-edge-detection-operators-in-digital-image-processing/](https://www.geeksforgeeks.org/image-edge-detection-operators-in-digital-image-processing/)
- [5] [old.upm.ro/facultati-departamente/stiinte-litere/conferinte/situl-integrare-europeana/Lucrari2/Ioan-Ispas.pdf](https://old.upm.ro/facultati-departamente/stiinte-litere/conferinte/situl-integrare-europeana/Lucrari2/Ioan-Ispas.pdf)
- [6] [perform.usv.ro/rapoarte/10/raport-cercetare-1.pdf](https://perform.usv.ro/rapoarte/10/raport-cercetare-1.pdf)
- [7] [rria.ici.ro/wp-content/uploads/2020/10/Complex-Architectures-Used-in-Image-Parallel-Processing.pdf](https://rria.ici.ro/wp-content/uploads/2020/10/Complex-Architectures-Used-in-Image-Parallel-Processing.pdf)
- [8] [www.miv.ro/books/MIvanovici-PI.pdf](https://www.miv.ro/books/MIvanovici-PI.pdf)
- [9] [researchgate.net/profile/Gaurav-Gupta-53/publication/Image-Filtering-Algorithms-and-Techniques-A-Review.pdf](https://researchgate.net/profile/Gaurav-Gupta-53/publication/Image-Filtering-Algorithms-and-Techniques-A-Review.pdf)
- [10] [books.google.ro/Theo-Pavlidis-Algorithms-for-Graphics-and-Image-Processing](https://books.google.ro/Theo-Pavlidis-Algorithms-for-Graphics-and-Image-Processing)
- [11] [homepages.inf.ed.ac.uk/rbf/HIPR2/histogram](https://homepages.inf.ed.ac.uk/rbf/HIPR2/histogram)