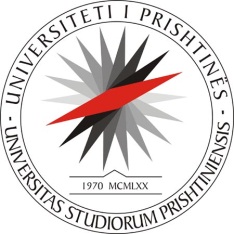
**UNIVERSITETI I PRISHTINËS “HASAN PRISHTINA”**

FAKULTETI I SHKENCAVE MATEMATIKO-NATYRORE

Departamenti i Matematikës

****

**Raporti për projektin e dytë**

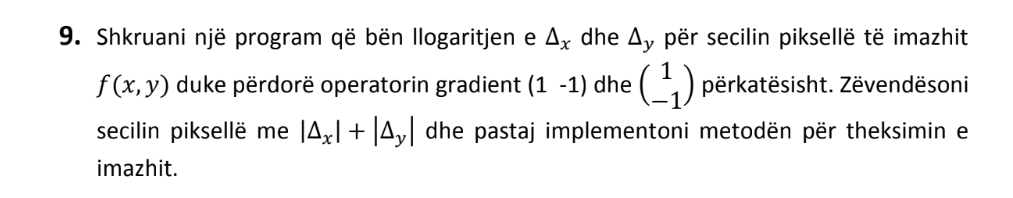
**LËNDA:** Procesim i Imazheve

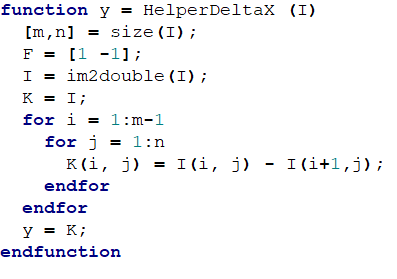
**Asistenti: Punoi:**

Besnik Duriqi Flamur Avdylaj

Arlind Arifaj

*Prishtinë, Maj 2023*



  
Për të zgjidhur detyrën e nëntë, është vepruar si në vijim: Fillimisht janë lexuar kushtet dhe përshkrimi i detyrës. Është analizuar detyra mirë dhe është shkarkuar fotoja "lena\_gaussian\_noise" që është e tipit TIF. Më pas, është filluar me realizimin e detyrës.  
Kërkesa e detyrës është qe ta ndërtojmë një program që llogarit vlerat e Δx dhe Δy për çdo piksel të një imazhi duke përdorur operatorët e gradientit (1 -1) dhe (1; -1) përkatësisht. Këto operatorë janë matrica të gradientit horizontal dhe vertikal.  
Δx llogaritet përmes aplikimit të operatorit të gradientit horizontal, që është një matricë e thjeshtë (1 -1). Kjo matricë përdoret për të marrë ndryshimin e intensitetit të ngjyrës në drejtim horizontal në të gjithë imazhin ndersa Δy llogaritet përmes aplikimit të operatorit të gradientit vertikal, që është një matricë e thjeshtë [1; -1].  
Pas llogaritjes së Δx dhe Δy, programi duhet të zëvendësojë secilin piksel në imazh me |Δx| + |Δy|. Kjo vlerë përfaqëson theksimin e imazhit, që është një masë e ndryshimit të intensitetit të ngjyrës për çdo piksel. Imazhi i theksuar do të shfaqë konturet dhe detajet e imazhit më qartë.  
  
Për realizimin e kësaj detyre, do të përdorim dy funksione ndihmëse: "HelperDeltaX" dhe "HelperDeltaY". Më poshtë do të paraqesim kodin për këto dy funksione dhe do t'i shpjegojmë ato  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
   
  
   
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
Kodi i funksionit **HelperDeltaX** llogarit ndryshimin e intensitetit të ngjyrës në drejtim horizontal (Δx) për çdo piksel të imazhit, po fillojme me komentimin e kodit per secilin rresht.  
**function y = HelperDeltaX (I)** : Ky është deklarimi i funksionit. Ky funksion merr një imazh të dhënë si hyrje (I) dhe llogarit ndryshimin e intensitetit të ngjyrës në drejtim horizontal (Δx) për çdo piksel të atij imazhi. Pas përpunimit, ai kthen një imazh të ri (y) që përmban vlerat e llogaritura të Δx për çdo piksel të imazhit të dhënë.  
**[m,n] = size(I);** :Ky rresht merr madhësinë e imazhit hyrës (I) dhe ruan numrin e rreshtave (m) dhe kolonave (n) për të përdorur më vonë në ciklet for.  
**F = [1 -1];**: Ky rresht krijon operatorin e gradientit horizontal (1 -1) që do të përdoret për të llogaritur Δx.  
**I = im2double(I);**: Ky rresht konverton imazhin hyrës në vlera të dyfishta (numra me presje dhjetore) për të siguruar që llogaritjet të jenë të saktë.

**K = I;:** Ky rresht krijon një kopje të imazhit hyrës që do të përdoret për të ruajtur vlerat e llogaritura të Δx për çdo piksel.

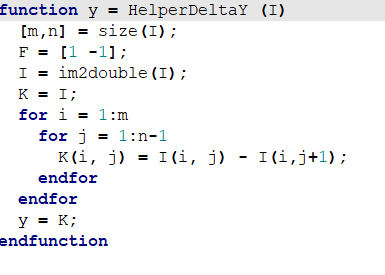
**for i = 1:m-1**: Ky cikel for përdoret për të iteruar nëpër rreshtat e imazhit. ‘ -1’ në ‘m-1’ është e nevojshme për të mos tejkaluar kufijtë e imazhit gjatë llogaritjes së Δx.

**for j = 1:n**: Ky cikel for përdoret për të iteruar nëpër kolonat e imazhit.

**K(i, j) = I(i, j) - I(i+1,j);:** Ky rresht llogarit ndryshimin e intensitetit të ngjyrës në drejtim horizontal për pikselin aktual dhe e vendos atë në imazhin e kopjuar K.

**endfor:** Këto janë mbylljet e cikleve for që përfshijnë rreshtat dhe kolonat e imazhit.

**y = K;:** Ky rresht i jep imazhit të kopjuar K, që përmban vlerat e llogaritura të Δx për çdo piksel, si rezultat i funksionit.

**endfunction**: Ky rresht përfundon funksionin HelperDeltaX.  
  
  
Tani do të paraqesim funksionin "HelperDeltaY" dhe do ta komentojmë atë rresht për rresht, duke marrë parasysh kërkesat e detyrës.  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
Funksioni HelperDeltaY llogarit ndryshimin e intensitetit të ngjyrës në drejtim vertikal (Δy) për çdo piksel të imazhit. Ky ndryshim tregon se sa ndryshon intensiteti i ngjyrës nga një piksel në pikselin e tij të djathtë. Ky funksion përdor një filter të gradientit [1 -1] për të llogaritur këtë ndryshim dhe kthen një imazh të ri me vlerat e Δy për çdo piksel. Kjo ndihmon në theksimin e kontureve dhe detajeve të imazhit fillestar.  
**function y = HelperDeltaY (I):** Ky është deklarimi i funksionit. Ky funksion merr një imazh të dhënë si hyrje (I) dhe llogarit ndryshimin e intensitetit të ngjyrës në drejtim vertikal (Δy) për çdo piksel të atij imazhi. Pas përpunimit, ai kthen një imazh të ri (y) që përmban vlerat e llogaritura të Δy për çdo piksel të imazhit të dhënë.  
**[m,n] = size(I):** Ky rresht merr madhësinë (numrin e rreshtave m dhe kolonave n) të imazhit të dhënë.  
**F = [1 -1]:** Ky rresht deklaron filterin e gradientit që do të përdoret për të llogaritur Δy. Në këtë rast, filteri është një matricë 1x2 me vlerat [1 -1].  
**I = im2double(I):** Ky rresht konverton imazhin e dhënë nga tipi i të dhënave të imazhit origjinal (p.sh., uint8) në një tip të dhënash me presje të dyfishtë, për të mundësuar llogaritje me presje të lartë.

**K = I:** Ky rresht krijon një kopje të imazhit të konvertuar I, e cila do të përdoret për të ruajtur rezultatet e llogaritur të Δy.

**for i = 1:m:** Ky rresht fillon një cikël që kalon nëpër çdo rresht të imazhit.

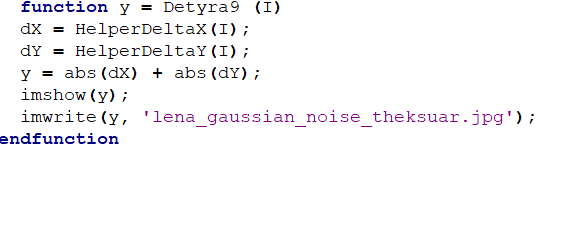
**for j = 1:n-1:** Ky rresht fillon një cikël të brendshëm që kalon nëpër çdo kolonë të imazhit, përveç kolonës së fundit.

**K(i, j) = I(i, j) - I(i, j+1):** Ky rresht llogarit ndryshimin e intensitetit të ngjyrës në drejtim vertikal (Δy) midis pikselit aktual dhe pikselit të tij të djathtë dhe e vendos atë në imazhin e kopjuar K.

**endfor:** Ky rresht përfundon ciklin e brendshëm që kalon nëpër kolonat e imazhit.

**endfor:** Ky rresht përfundon ciklin e jashtëm që kalon nëpër rreshtat e imazhit.

**y = K:** Ky rresht cakton imazhin e përpunuar K si rezultat i kthyer nga funksioni, i cili përmban vlerat e llogaritura të Δy për çdo piksel të imazhit.  
**endfunction**: Ky rresht përfundon funksionin HelperDeltaX.

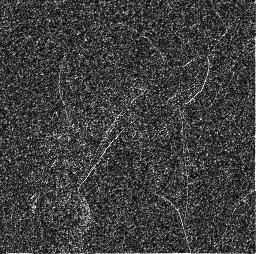
Tani po e paraqesim funksionin kryesor:  
  
Funksioni Detyra9 është funksioni kryesor që koordinon përpunimin e imazhit. Ky funksion merr imazhin hyrës (I) dhe përdor dy funksionet ndihmëse, HelperDeltaX dhe HelperDeltaY, për të llogaritur ndryshimet e intensitetit të ngjyrës në drejtim horizontal (Δx) dhe vertikal (Δy) për çdo piksel.

**dX = HelperDeltaX(I);:** Llogarit ndryshimin e intensitetit të ngjyrës në drejtim horizontal (Δx) për çdo piksel të imazhit të dhënë duke përdorur funksionin ndihmës HelperDeltaX.

**dY = HelperDeltaY(I);:** Llogarit ndryshimin e intensitetit të ngjyrës në drejtim vertikal (Δy) për çdo piksel të imazhit të dhënë duke përdorur funksionin ndihmës HelperDeltaY.

**y = abs(dX) + abs(dY);:** Zëvendëson çdo piksel të imazhit me vlerën absolute të Δx dhe Δy për të krijuar një imazh të theksuar.

**imshow(y);:** Shfaq imazhin e theksuar në dritare të veçantë të vizualizimit.

**imwrite(y, 'lena\_gaussian\_noise\_theksuar.jpg');:** Ruaj imazhin e theksuar në formatin  
e caktuar të skedarit (në këtë rast, .jpg) me emrin e dhënë lena\_gaussian\_noise\_theksuar.jpg   
  
Për të parë rezultatet përfundimtare duke përdorur këto funksione, kemi vepruar fillimisht duke krijuar "Octave files" për secilën nga funksionet e lartë përmendura dhe i kemi ruajtur secilën prej tyre, si dhe foton të cilën e kemi shkarkuar nga email-i i dhënë me kërkesa. Të gjitha këto i kemi vendosur në një skedar. Për të ekzekutuar këto funksione, kemi hapur programin Octave, dhe në dritaren e komandave (command window) kemi ndërruar direktorinë ku janë ruajtur skriptat e Octave (Detyra9.m, HelperDeltaX.m dhe HelperDeltaY.m) dhe imazhi (lena\_gaussian\_noise.tif). Për ta bërë këtë, kemi përdorur komandën:  
dhe pastaj për të parë rezultatet kemi përdorur komandat:  
  
Dhe tani po e paraqesim foton origjinale dhe ndryshimet qe janë bërë  
  
**Fig. 1. Rezultati i fituar nga detyra e nëntë**

**Referencat**[1].Gonzalez, R.C., Woods, R.E., & Eddins, S.L. (2009). Digital Image Processing Using MATLAB (2nd ed.). Gatesmark Publishing.  
[2]. Wikipedia: Image gradient.