**UNIVERSITETI I PRISHTINËS**

**FAKULTETI I SHKENCAVE MATEMATIKE NATYRORE**

DEPARTAMENTI I MATEMATIKËS

Programi: Shkencë Kompjuterike



Lënda: Procesim i imazheve

Detyra e dytë

Profesori: Studenët:

Besnik Duriqi Erëza Asllani

Fatjonë Thaci

Fjolla Selimi

Maj 2023

**Përmbajtja**

[**Lista e figurave** 2](#_Toc135692522)

[**Përshkrimi i detyrës** 3](#_Toc135692523)

[**Nëndetyra e parë** 3](#_Toc135692524)

[**Nëndetyra e dytë** 5](#_Toc135692525)

[**Nëndetyra e tretë** 6](#_Toc135692526)

[**Nëndetyra e katërt** 8](#_Toc135692527)

# **Lista e figurave**

[Figura 1. Zhurma Rayleigh. 5](#_Toc135692484)

[Figura 2. Imazhet pas aplikimit të zhurmës dhe heqjes së saj. 6](#_Toc135692485)

[Figura 3. Aplikimi i filterit Hg. 8](#_Toc135692486)

[Figura 4. Figurat 2 dhe 3 në detyrë. 8](#_Toc135692487)

[Figura 6. Gjetja e elementit strukturor. 9](#_Toc135692488)

# **Përshkrimi i detyrës**

Detyra përfshin 4 nëndetyra tjera.

Nëndetyra e parë është krijimi i funksionit ‘zhurma’ që gjeneron zhurmën me një shpërndarje të Rayleigh, Eksponenciale, LogNormal, Salt&Pepper dhe Erlang. Ky funksion zhurma(‘Shpërndarja’, x ,y ,z ,k ,s ,a ,b) merr disa parametra hyrës: shpërndarjen me të cilën gjenerohet zhurma, x- rreshti prej ku fillon zhurma, y-shtylla prej ku fillon zhurma, z-rreshti ku përfundon zhurma, k- shtylla ku përfundon zhurma, s- nr. i shtresave (1, 2 ose 3) dhe parametrat a dhe b.

Nëndetyra e dytë është shtimi i zhurmës periodike te imazhi ‘lena.tiff’ dhe largii i saj përmes transformimeve të Furie.

Nëndetyra e tretë është krijmi i një filteri që largon frekuencat e përqendruara rreth pikave të caktuara.

Nëndetyra e katërt është gjetja e elementit strukturor dhe operacioneve nga matematika morfologjike që japin rezulatin në një figurë.

# **Nëndetyra e parë**

Sic u përshkrua më lartë, kjo nëndetyrë përfshin implementimin e një funksioni që gjeneron zhurmën me ndonjërën nga shpërndarjet Rayleigh, Eksponenciale, LogNormal, Salt&Pepper dhe Erlang.

Në Octave implementimi i këtij funksioni është bërë me anë të kodit:

function zhurma(Shperndarja, x, y, z, k, s, a, b)

%Krijimi i nr te rreshtave dhe te kolonave

nrRr = abs(z - x) + 1;

nrK = abs(k - y) + 1;

%Krijimi i matrices se zhurmes me numer nrRr te rreshtave dhe nrK te kolonave

zhurma = zeros(nrRr, nrK, s);

%Zaktimi i zhurmes ne baze te shperndarjes

switch Shperndarja

case 'Rayleigh'

for i = 1:s

zhurma(:,:,i) = a + b \* sqrt(-2 \* log(1 - rand(nrRr, nrK)));

end

case 'Eksponenciale'

for i = 1:s

zhurma(:,:,i) = exprnd(1/a, nrRr, nrK);

end

case 'LogNormal'

for i = 1:s

zhurma(:,:,i) = exp(b \* randn(nrRr, nrK) + a);

end

case 'Salt&Pepper'

for i = 1:s

zhurma(:,:,i) = SaltPepper(nrRr, nrK, a);

end

case 'Erlang'

for i = 1:s

zhurma(:,:,i) = -a \* sum(log(1 - rand(nrRr, nrK)), 3);

end

otherwise

error('Shpërndarja qe keni zgjedhur nuk eshte e vlefshme.');

end

%Shfaqja e zhurmes

disp(zhurma);

end

%funksioni ndihmes per Salt&Pepper

function sp = SaltPepper(x, y, p)

sp = zeros(x, y);

sp(rand(x, y) < p/2) = -1;

sp(rand(x, y) >= 1 - p/2) = 1;

end

Sic shihet, për përcaktimit e shpërndarjes kei përdorur switch dhe brenda secilit case kemi iplementuar funksionin zhurma për shpërndarjen e caktuar.

Një shembull i tillë në Octave mund të përdoret me anë të kodit:

>>zhurma('Rayleigh', 1, 1, 5, 5, 3, 2, 3);

Outputi i kësaj komande është ky:

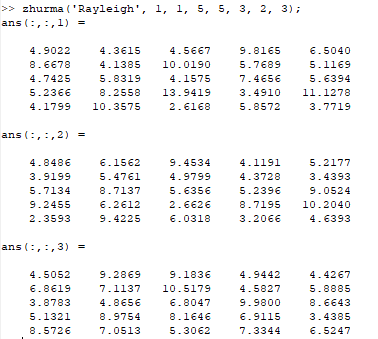


Figura 1. Zhurma Rayleigh.

Pra, këtu kemi thirrur funksionit zhurma për shpërndarjen Rayleigh i cili fillon nga rreshti dhe shtylla e parë dhe vazhdon deri te rreshti dhe shtylla e pestë me tri shtresa si dhe parametrat a = 2 dhe b = 3.

# **Nëndetyra e dytë**

Nëndetyra e dytë kërkon që të shtohet zhurma periodike (1) në imazhin ‘lena.tiff’ dhe pastaj të largohet me anë të transformimeve Furie.

(1)

Kodi në Octave për këtë detyrë është ky:

>> pkg load image

>> lena = imread('C:\Users\HP\Desktop\UP-Fjolla\Semestri 6\Procesimi i imazheve\Detyra 3\lena.tiff');

>> lena = im2double(lena\_img);

>> [x, y] = meshgrid(1:size(lena, 2), 1:size(lena, 1));

>> zhurma\_periodike = 22 \* cos((pi \* x) / 3 + (pi \* y) / 3) + 27 \* sin((pi \* x) / 2 + (pi \* y) / 3);

>> zhurma\_periodike = (zhurma\_periodike - min(zhurma\_periodike(:))) / (max(zhurma\_periodike(:)) - min(zhurma\_periodike(:)));

>> lena\_zhurmeP = lena + zhurma\_periodike;

>> lena\_frTrans = fft2(lena\_zhurmeP);

>> threshold = 100;

>> lena\_frTrans(abs(lena\_frTrans) > threshold) = 0;

>> lena\_paZhurme = real(ifft2(lena\_frTrans));

>> subplot(1, 3, 1);

>> imshow(lena);

>> title('Imazhi Origjinal');

>> subplot(1, 3, 2);

>> imshow(lena\_zhurmeP);

>> title('Imazhi me zhurme');

>> subplot(1, 3, 3);

>> imshow(lena\_paZhurme);

>> title('Imazhi pas heqjes se zhurmes');

Pra, së pari lexojmë imazhin ‘lena.tiff” dhe pastaj e krijojmë një meshgrid për koordinatat x dhe y të imazhit. Pastaj, me anë të formulës së dhënë krijojmë zhurmën periodike dhe e normalizojmë atë. E shtojmë këtë zhurmë në imazhin ‘lena.tiff’ dhe pastaj llogarisim transformimin 2D Furier dhe e përdorim për heqjen e zhurmës periodike të vendosur më herët.

Si rezultat, shfaqim një subpot me 3 kolona ku në kolonën e parë shfaqet imazhi origjinal, në të dytën imazhi me zhurmë periodike dhe në të tretën imazhi pas heqjes së zhurmës periodike me anë të tranformimin 2D Furier.

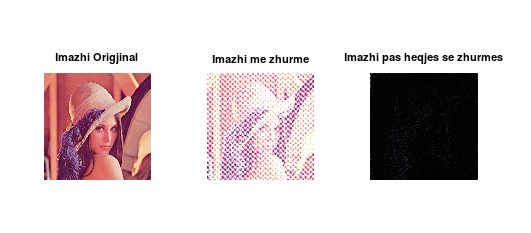


Figura 2. Imazhet pas aplikimit të zhurmës dhe heqjes së saj.

# **Nëndetyra e tretë**

Nëndetyra e tretë kërkon krijimin e filterit :

ku D0 = (u0, v0) është qendra e njërës nga frekuencat që duhet të largohet.

D1 dhe D2 llogariten me anë të formulës:

ku i =1, 2 dhe M, N madhësia e imazhit.

Për të krijuar këtë filtër në Octave së pari krijojmë funksionin Hg që krijon filterin e kërkuar:

function hg = Hg(M, N, u0, v0, Do)

%Krijon nje meshgrid per koordinatat x dhe y te imazhit

[u, v] = meshgrid(1:N, 1:M);

%Vlerat e llogaritura te D1 DHE D2 me ane te formulave perkatese

D1 = sqrt((u - M/2 + ((-1)^1) \* u0).^2 + (v - N/2 + ((-1)^1) \* v0).^2);

D2 = sqrt((u - M/2 + ((-1)^2) \* u0).^2 + (v - N/2 + ((-1)^2) \* v0).^2);

%Llogarit filterin me ane te formules se dhene

hg = 1 - exp(-0.5 \* ((D1 .\* D2) / Do^2));

end

Pastaj, për të përdorur këtë filter në imazhin ‘lena.tiff’ përdorim këto komanda në Octave:

>> pkg load image

>> lena = imread('C:\Users\HP\Desktop\UP-Fjolla\Semestri 6\Procesimi i imazheve\Detyra 3\lena.tiff');

>> lena = im2double(lena);

>> [M, N, ~] = size(lena);

>> u0 = 30;

>> v0 = 30;

>> Do = 100;

>> lena\_frTrans = fft2(lena);

>> hg = Hg(M, N, u0, v0, Do);

>> lena\_frTrans\_filt = lena\_frTrans .\* hg;

>> lena\_paFilter = real(ifft2(lena\_frTrans\_filt));

>> subplot(1, 3, 1);

>> imshow(lena);

>> title('Imazhi origjinal');

>> subplot(1, 3, 2);

>> imshow(abs(fftshift(hg)));

>> title('Filteri');

>> subplot(1, 3, 3);

>> imshow(lena\_paFilter);

>> title('Imazhi pas hjekejes se filterit');

Së pari, loadojmë paketën e nevojshme, lexojmë imazhin dhe marrim madhësinë e imazhit. Vendosim vlerat për D0, u0, v0 sipas dëshirës. Aplikojmë transformimin 2D Furier në imazhin lena dhe pastaj llogarisim filterin Hg. Aplikojmë filterin Hg në imazhin e transformuar dhe në fund e pastrojmë përsëri imazhin nga filteri. Si përfundim, shfaqim imazhet para aplikimit të filterit dhe pas hjekes së tij.

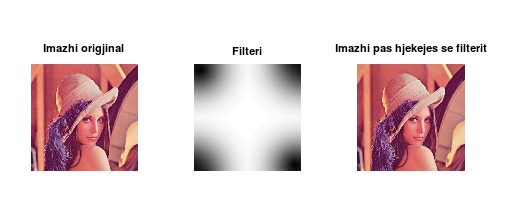
****

Figura 3. Aplikimi i filterit Hg.

# **Nëndetyra e katërt**

Nëndetyra e katërt kërkon që të gjendet elementi strukturor dhe operacionet e matematikës morfologjike që do të prodhojnë rezultatin e Figurës 3 duke vepruar mbi regjionin binar të dhënë në Figurën 2.

Figura 4. Figurat 2 dhe 3 në detyrë.

Kodi në Octave për të ardhur te ky rezultat është:

pkg load image

% Lexon imazhin

image = imread('C:\Users\HP\Desktop\UP-Fjolla\Semestri 6\Procesimi i imazheve\Detyra 3\figura 2.png');

% Kthen ne imazh binar bardh dhe zi

binaryImage = im2bw(image);

% Definon elementin strukturor

se = strel('square', 50); % Elementi strukturor square me madhesi 3x3

% Aplikon dilation

dilatedImage = imdilate(binaryImage, se);

% Shfaq imazhin origjinal dhe pas dilation

subplot(1, 2, 1);

imshow(binaryImage);

title('Original Image');

subplot(1, 2, 2);

imshow(dilatedImage);

title('Dilated Image');

Rezultati është ky:

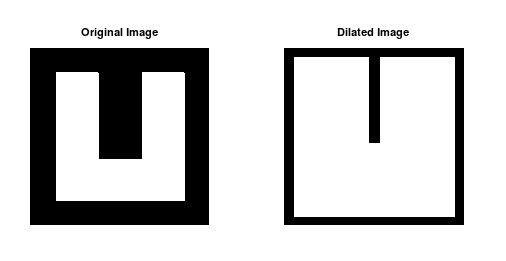


Figura 6. Gjetja e elementit strukturor.