

UNIVERSITETI I PRISHTINËS
FAKULTETI I SHKENCAVE MATEMATIKORE
DHE NATYRORE
DEPARTAMENTI I MATEMATIKËS

Programi: *Shkencë Kompjuterike*



LËNDA: Procesimi i imazheve

TEMA: Përpunimi i imazhit përmes ndarjes në blloqe.

Profesori:

Artan Berisha

Besnik Duriqi

Studenti:

Erlisa Lokaj

Ermira Haziri

Erion Hoti

Prill 2023

Përmbajtja

<i>Abstrakt</i>	<i>3</i>
<i>Hyrje.....</i>	<i>3</i>
<i>Përshkrimi i metrikave</i>	<i>3</i>
<i>Përshkrimi i skriptës.....</i>	<i>4</i>
<i>Përshkrimi i funksioneve</i>	<i>7</i>
<i>Rezultatet.....</i>	<i>8</i>
<i>Përfundim</i>	<i>10</i>

Abstrakt

Ky projekt është një shembull i përpunimit të imazheve, ku përdoren matricat e blloqeve të madhësive të ndryshme për të llogaritur disa metrika si: vlera mesatare, devijimi standard, maksimumi, minimumi, mediana dhe energjia. Përmes këtyre, rezultatet janë shfaqur në imazhe të veçanta dhe janë analizuar për të kuptuar ndryshimet midis blloqeve të ndryshme dhe tipareve të përgjithshme të imazhit. Kjo është një metodë e përdorur shpesh në procesimin e imazheve, dhe ky projekt mund të shërbejë si një udhëzues për të kuptuar më mirë si funksionojnë metodat e përpunimit të imazheve në praktikë.

Hyrje

Në epokën e sotme dixhitale, përdorimi i përpunimit të imazhit dhe vizionit kompjuterik është bërë gjithnjë e më i përhapur në fusha të ndryshme si mjekësia, inxhinieria dhe argëtimi. Një nga hapat themelorë në përpunimin e imazhit është ndarja e imazhit në blloqe për analiza të mëtejshme. Megjithatë, zgjedhja e madhësisë së bllokut luan një rol vendimtar në saktësinë dhe besueshmërinë e rezultateve të marra nga përpunimi i imazhit. Prandaj, kuptimi i karakteristikave të imazhit, si niveli i detajeve, kontrasti dhe ndryshueshmëria, është thelbësor në përcaktimin e madhësisë së duhur të bllokut për një aplikacion të caktuar.

Në këtë raport, ne do të eksplorojmë efektet e madhësisë së bllokut në përpunimin e imazhit duke analizuar metrika të ndryshme si vlerat mesatare, devijimi standard, vlerat minimale dhe maksimale dhe shpërndarja e energjisë. Për të arritur këtë qëllim, ne kemi përdorur softuerin Octave, i cili ofron një sërë funksionesh matematikore për analiza numerike, matricore dhe sinjale. Përveç kësaj, Octave ofron funksione të ndryshme të vizualizimit të imazhit që na lejojnë të paraqesim rezultatet tona në një mënyrë të qartë dhe koncize.

Kërkesat e detyrës fillesatare përshkruajnë problemin në këtë formë:

Imazhi lena.tif (256 x 256) duhet të ndahet në blloqe të madhësisë 2×2 . Në secilin prej këtyre blloqeve llogaritni vlerën mesatare dhe devijimin standard, më pas ruani këto rezultate në dy imazhe përkatësisht. Vizualizoni imazhet dhe interpretoni rezultatet.

Mirëpo ne do të shqyrtojmë rezultatet e marra duke ndryshuar madhësitë e bllokut nga 2×2 në 8×8 dhe do të vëzhgojmë se si ndryshimet në madhësinë e bllokut ndikojnë në qartësinë, detajet dhe interpretueshmërinë e imazheve. Për më tepër, ne do të diskutojmë ndikimin e madhësisë së bllokut në kërkesat llogaritëse të algoritmeve dhe shkëmbimin midis saktësisë dhe kohës së përpunimit.

Në përgjithësi, ky raport synon të japë njohuri mbi rëndësinë e zgjedhjes së një madhësie të përshtatshme blloku për detyrat e përpunimit të imazhit dhe të ndihmojë studiuesit dhe praktikuesit të marrin vendime të informuara kur zgjedhin madhësitë e bllokut për aplikimet e tyre specifike.

Përshkrimi i metrikave

Metrikat e llogaritura janë përdorur për të analizuar disa karakteristika të imazhit. Në këtë projekt, janë llogaritur mesatarja, devijimi standard, minimumi, maksimumi, mediana dhe energjia për blloqeve të madhësive të ndryshme të imazhit.

Mesatarja, siç thuhet edhe në emër, tregon vlerën mesatare të një sërë numrash. Në këtë projekt, mesatarja është llogaritur për secilin kanal RGB të secilit bllok të imazhit. Të gjitha vlerat në bllok shumëzohen dhe pastaj pjesëtohen me numrin e elementeve në bllok. Një mesatare e lartë tregon

se vlerat në bllok janë në përgjithësi të larta. Kjo na tregon vlerën mesatare të ngjyrave në secilin bllok, dhe mund të na ndihmojë të kuptojmë se si ndryshojnë ngjyrat brenda dhe jashtë secilit bllok.

Devijimi standard tregon sa shumë vlerat e një serie janë larg mesatares. Në këtë projekt, devijimi standard është llogaritur për secilin kanal RGB të secilit bllok të imazhit. Nëse devijimi standard është i madh, atëherë vlerat në bllok janë më të shpërndara dhe më pak të konsistentë. Nëse devijimi standard është i vogël, atëherë vlerat në bllok janë më të konsistenta dhe të ngjashme me njëra-tjetrën. Kjo na tregon se sa shumë ngjyra ndryshojnë në secilin bllok dhe nëse kjo ndryshim është i rëndësishëm në krahasim me mesataren.

Minimumi dhe maksimumi tregojnë vlerat minimale dhe maksimale të një sërë numrash. Në këtë projekt, minimumi dhe maksimumi janë llogaritur për secilin kanal RGB të secilit bllok të imazhit. Kjo na tregon se cilat ngjyra janë më të errëta dhe më të ndritshme në secilin bllok. Një vlerë minimale e ulët tregon se vlerat në bllok janë më të ulëta ndërsa një vlerë maksimale e lartë tregon se vlerat në bllok janë më të larta.

Mediana është vlera që ndodhet në mes të një sërë numrash, kur të rendisim nga më e vogla në më të madhe. Në këtë projekt, mediana është llogaritur për secilin kanal RGB të secilit bllok të imazhit. Kjo na tregon vlerën mesatare të ngjyrave, duke injoruar ndikimin e vlerave të skajshme që mund të kenë ndryshime të mëdha në mesatare. Nëse ka shumë vlera të ekstremiteteve në bllok (p.sh., disa vlera shumë të larta ose të ulëta), mesatarja mund të jetë e ndikuar shumë nga këto vlera. Në këtë rast, mediana është një metrikë më e mirë për të përshkruar vlerat e bllokut.

Energjia është shuma e vlerave të kuadratuara të një sërë numrash. Në këtë projekt, energjia është llogaritur për secilin kanal RGB të secilit bllok të imazhit. Kjo na tregon se sa shumë energji ekziston në secilin bllok, dhe mund të na ndihmojë të kuptojmë se cilat pjesë të imazhit janë më të ngjeshura me ngjyra. Energjia është e lartë nëse ka shumë vlera të larta në bllok.

Përshkrimi i skriptës

Skripta e krijuar për realizimin e detyrës përfshin disa hapa dhe funksione të cilat do të përshkruhen me rradhë në vazhdim.

```
clear all; %Fshi të gjitha variablat nga memory.
pkg load image % Ngarko paketën "image" në Octave.
%Ngarko imazhin e quajtur "lena.tiff" dhe e konverto në formatin double.
img = im2double(imread("/Users/erlisalokaj/Desktop/Projekti 2 Image
processing/lena.tiff"));
img_size = size(img);
```

Rreshti i parë, "clear", fshin të gjitha variablat nga memoria për të siguruar që nuk ka konflikte me kodin e mëparshëm. Është një teknikë e përdorur shpesh në Octave. Rreshti i dytë, "pkg load image", ngarkon "imazhin" e paketës së përpunimit të imazhit në Octave, e cila ofron disa funksione dhe mjete për të punuar me imazhe. Rreshti i tretë, lexon imazhin me emrin "lena.tiff" i vendosur në shtegun e specifikuar të skedarit dhe e konverton atë në një varg me precizion double. Imazhi që rezulton ruhet në variablën "img", ndërsa me variablën "img_size" ruajmë dimensionet e imazhit.

```
% Cakto dimensionet e secilit bllok si 2x2.
```

```
%block_size = [2 2];
% Kërko përdoruesit të jap një dimension të blloqeve
block_size = input("Shkruaj dimensionet e blloqeve: [rows cols]: ");
% Llogarit numrin e blloqeve në secilën drejtim.
num_blocks = floor(img_size(1:2) ./ block_size);
```

Skripti vendos përmasat e çdo blloku në 2x2 fillimisht, por më vonë e kërkon përdoruesin të vendosë madhësinë e bllokut. Numri i blloqeve në secilin drejtim llogaritet më pas duke pjesëtuar madhësinë e figurës me madhësinë e bllokut dhe duke rrumbullakosur poshtë në numrin e plotë më të afërt duke përdorur funksionin "floor". Kjo përcakton numrin e blloqeve që nuk mbivendosen që do të nxirren nga imazhi.

```
% Llogarit dimensionet e imazhit të pad-uar.
padded_size = num_blocks .* block_size;
% Krijë një matricë të re me vlera zero për të vendosur imazhin e pad-uar.
padded_img = zeros(padded_size(1), padded_size(2), size(img, 3));
% Vendos imazhin e vërtetë brenda matricës së pad-uar.
padded_img(1:size(img,1), 1:size(img,2), :) = img;
```

Këto rreshta kodi llogaritin dimensionet e imazhit të mbushur bazuar në madhësinë e bllokut dhe numrin e blloqeve. Pastaj, krijohet një matricë e re zerosh me madhësinë e imazhit të mbushur për të mbajtur imazhin e mbushur. Më në fund, imazhi origjinal vendoset brenda matricës së mbushur në këndin e sipërm majtas. Ky proces siguron që imazhi të mbushet siç duhet në madhësinë e kërkuar për hapin e përpunimit të bllokut.

```
% Ndërro formën e matricës së pad-uar në forma blloqesh.
blocks = reshape(padded_img, block_size(1), num_blocks(1),
block_size(2), num_blocks(2), 3);
%Ndërro renditjen e dimensioneve në matricën e blloqeve.
blocks = permute(blocks, [1 3 2 4 5]);
%Ndërro formën e matricës së blloqeve nga 4D në 2D.
blocks = reshape(blocks, block_size(1)*block_size(2), [], 3);
```

Këto rreshta kodi janë përgjegjëse për konvertimin e imazhit të mbushur në një grup blloqesh 3D. Rreshti i parë i kësaj pjese përdor funksionin e riformësimit për të riformuar imazhin e mbushur në blloqe të madhësisë së block_size, ku num_blocks është numri i blloqeve në çdo drejtim. Vargu që rezulton ka përmasat block_size (1) x num_blocks (1) x block_size (2) x num_blocks (2) x 3, ku dimensionin e fundit përfaqëson kanalet e ngjyrave RGB.

Rreshti pasues përdor funksionin permute për të riorganizuar dimensionet e grupit të bllokut për të pasur kanalet e ngjyrave si dimensionin e parë, të ndjekur nga dy dimensionet që përfaqësojnë madhësinë e bllokut dhe dy dimensionet që përfaqësojnë numrin e blloqeve në çdo drejtim. Vargu që rezulton ka përmasa 3 x madhësia e bllokut (1) x madhësia e bllokut (2) x numri_bloqet (1) x numri_bloqet (2).

Rreshti i fundit i pjesës së kodit përdor përsëri funksionin e riformësimit për të kthyer grupin 4D të blloqeve në një grup 2D, ku çdo rresht përfaqëson një bllok dhe tre kolonat përfaqësojnë kanalet

e ngjyrave. Vargu që rezulton ka përmasat $\text{blook_madhësia}(1) \times \text{madhësia_blloku}(2) \times (\text{numri_blloqe}(1) \times \text{numri_blloqe}(2)) \times 3$.

```
% Llogarit mesataren, devijimin standard, minimumin, maksimumin,  
medianën dhe energjinë e secilit bllok  
means = mean(blocks);  
stds = std(blocks);  
mins = min(blocks);  
maxs = max(blocks);  
meds = median(blocks);  
energy = sum(blocks.^2);
```

Seksioni tjetër i kodit llogarit statistika të ndryshme për çdo bllok të imazhit. Për të arritur këtë, blloqet 2D fillimisht konvertohen në vektorë 1D duke përdorur funksionin e riformësimit. Funksionet mesatare, std, min, max, median dhe shuma përdoren më pas për të llogaritur statistikat përkatëse për çdo bllok.

Vlen të përmendet se DCT (Transformimi Diskret i Kosinusit) zakonisht përdoret si një metodë më efikase për llogaritjen e këtyre statistikave. DCT është një lloj transformimi Furier që përdoret zakonisht në kompresimin e imazheve dhe videove. Ai funksionon duke transformuar një bllok vlerash pikselësh në një grup koeficientësh të frekuencës, të cilat më pas mund të kompresohen në mënyrë më efikase duke hedhur poshtë koeficientët me madhësi të vogla. Megjithatë, ky kod nuk përdor DCT, dhe përkundrazi llogarit statistikat drejtpërdrejt nga vlerat e pikselit në çdo bllok.

```
% Riformëso rezultatet në imazhe  
mean_img = reshape(means, num_blocks(1), num_blocks(2), []);  
std_img = reshape(stds, num_blocks(1), num_blocks(2), []);  
min_img = reshape(mins, num_blocks(1), num_blocks(2), []);  
max_img = reshape(maxs, num_blocks(1), num_blocks(2), []);  
med_img = reshape(meds, num_blocks(1), num_blocks(2), []);  
energy_img = reshape(energy, num_blocks(1), num_blocks(2), []);
```

Këto rreshta kodi po riformësojnë statistikat e bllokut të llogaritur më parë në imazhe për vizualizim dhe analizë më të lehtë. Çdo rresht kodi riformëson një nga statistikat në një imazh 2D me dimensione të barabarta me numrin e blloqeve në çdo drejtim. Për shembull, `mean_img` është një imazh 2D ku çdo piksel përfaqëson vlerën mesatare të koeficientëve DCT në një bllok përkatës të imazhit origjinal. Në mënyrë të ngjashme, `std_img` është një imazh ku çdo piksel përfaqëson devijimin standard të koeficientëve DCT në një bllok përkatës. E njëjta gjë vlen edhe për statistikat e tjera të llogaritura.

```
% Paaraqitja e imazheve përfundimtare përmes figures dhe subplot  
figure;  
subplot(2,3,1);  
imshow(imresize(mean_img, [256 256]));  
title("Mean values");  
subplot(2,3,2);  
imshow(imresize(std_img, [256 256]));  
title("Standard deviations");  
subplot(2,3,3);
```

```

imshow(imresize(min_img, [256 256]));
title("Minimum values");
subplot(2,3,4);
imshow(imresize(max_img, [256 256]));
title("Maximum values");
subplot(2,3,5);
imshow(imresize(med_img, [256 256]));
title("Median values");
subplot(2,3,6);
imshow(imresize(energy_img, [256 256]));
title("Energy");
waitfor(gcf);

```

Pjesa e fundit e skriptës është përgjegjëse për vizualizimin e rezultateve në një figurë me subplota të shumta. Krijon një figurë të re dhe më pas shton gjashtë subplota, secila me një titull të ndryshëm që korrespondon me llojin e statistikës që shfaqet. Funkzioni `imresize` përdoret për të ndryshuar madhësinë e çdo imazhi në një madhësi standarde prej 256x256 piksele përpara se të shfaqet në subplota. Funkzioni `waitfor(gcf)` përdoret për të pritur që përdoruesi të mbyllë figurën përpara se të vazhdojë me pjesën tjetër të kodit. Kjo i lejon përdoruesit të shikojë dhe analizojë rezultatet përpara se të vazhdojë.

Përshkrimi i funksioneve

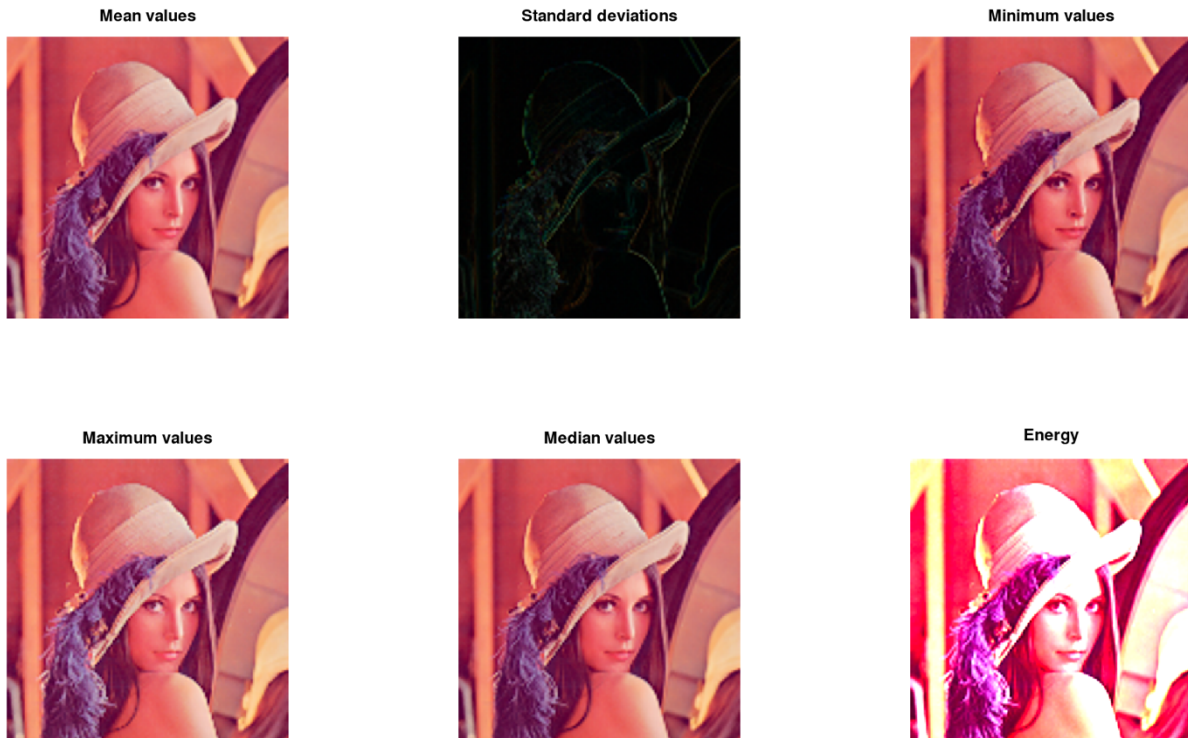
Skripta e realizur për problemin e dhënë përdor funksionet e mëposhtme:

clear	fshin të gjitha variablat nga memoria.
pkg load image	ngarkon paketën e imazhit në Octave.
imread	lexon një skedar imazhi nga disku.
im2double	konverton një imazh në format me saktësi të dyfishtë.
input	e kërkon përdoruesin të fusë vlerat.
floor	rrumbullakos një numër në numrin e plotë më të afërt.
zeros	krijon një matricë me zero.
reshape	riformëson një grup në një madhësi ose formë të re.
permute	permuton dimensionet e një vargu.
mean	njihson mesataren e një grupi.
std	njihson devijimin standard të një grupi.
min	gjen vlerën minimale të një vargu.
max	gjen vlerën minimale të një vargu.
median	njihson medianën e një grupi.
sum	njihson shumën e një vargu.
imshow	shfaq një imazh.
imresize	ndryshon përmasat e një imazhi.
title	përcakton titullin e një komploti.
figure	krijon një dritare të re figurash.
subplot	krijon një subplot brenda një figure.
waitfor	pret që një figurë të mbyllet.

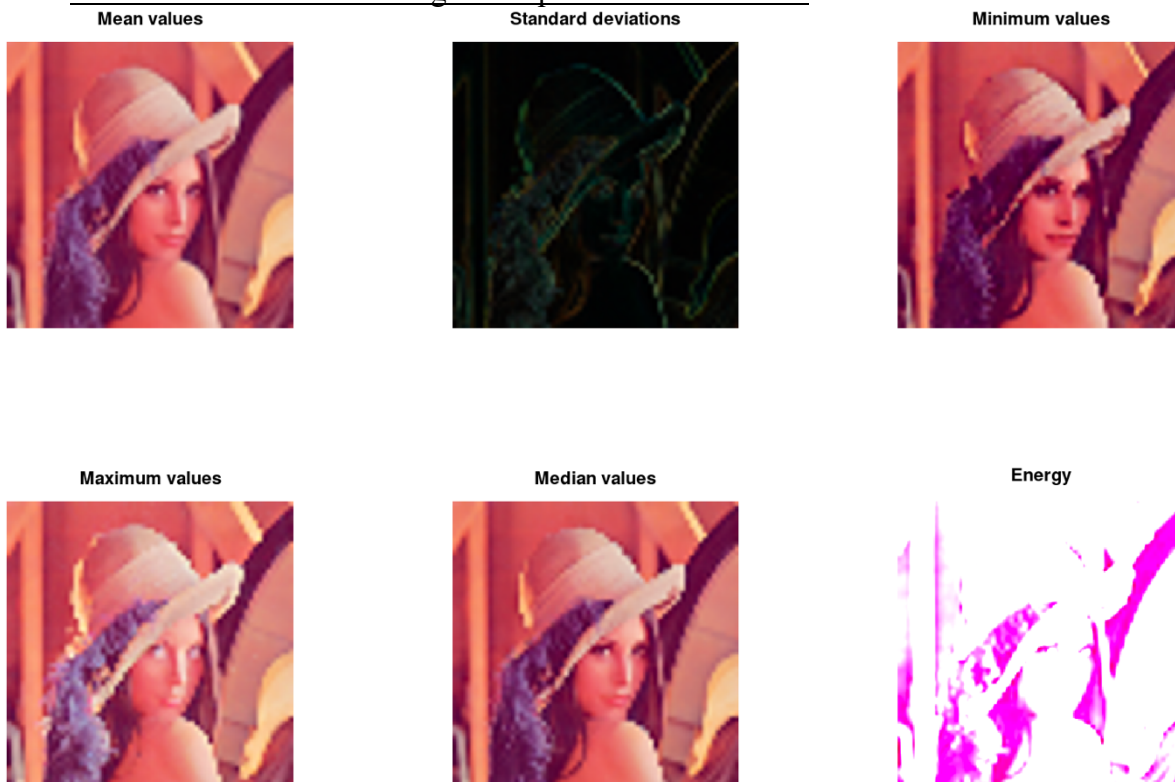
Rezultatet

Problemi i dhënë kërkon ndarjen e imazhit të paracaktuar në blloqe të mprehta të madhësisë 2×2 , mirëpo ne kemi mundësuar që përdoruesi të caktoj dimensionet sipas dëshirës për sa kohë që përshtaten me dimensionet 256×256 të imazhit. Kështu, do të tregojmë rezultatet e metrikave të llogaritura për dimensionet 2×2 , 4×4 dhe 8×8 .

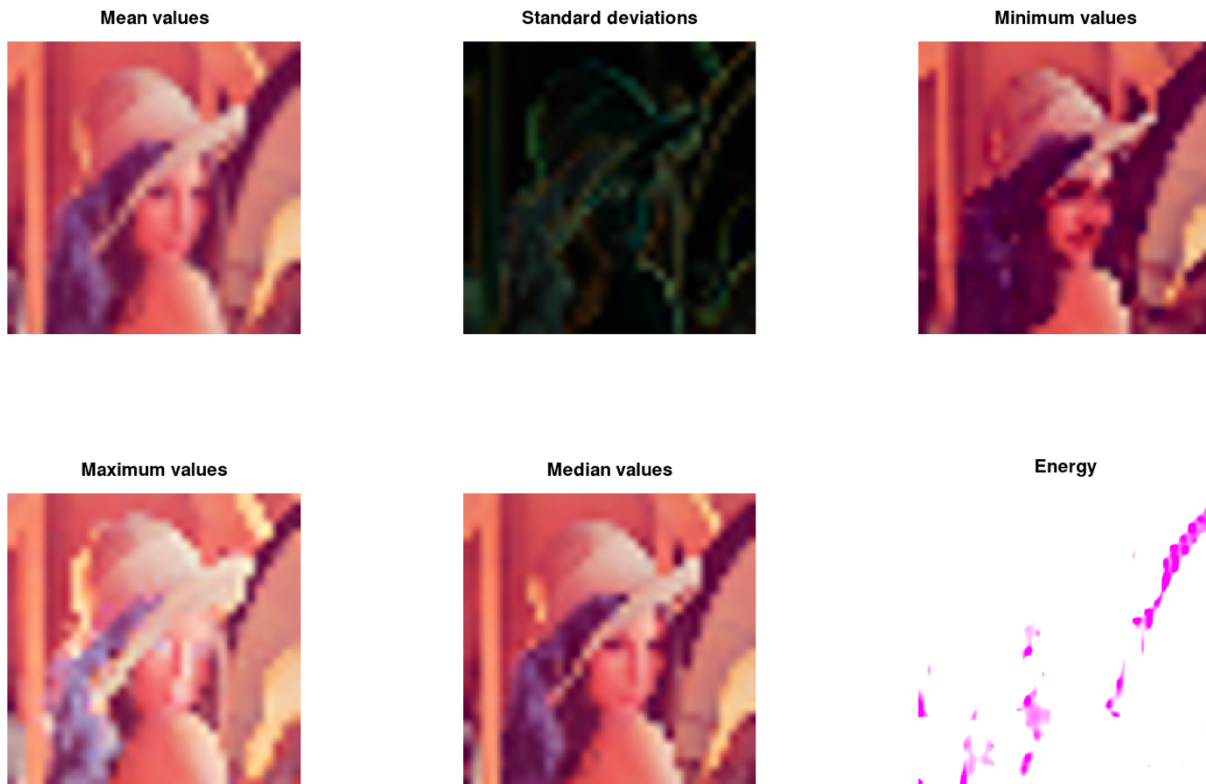
- Rezultatet e metrikave të llogaritura për dimensionet 2×2 :



- Rezultatet e metrikave të llogaritura për dimensionet 4×4 :



- Rezultatet e metrikave të llogaritura për dimensionet 8x8:



Në procesin e ndarjes së imazhit në blloqe dhe llogaritjen e metrikave të ndryshme, një aspekt shumë i rëndësishëm është zgjedhja e madhësisë së bllokut. Siç shihet edhe nga rezultatet e paraqitura, madhësia e bllokut ndikon në paraqitjen dhe informacionin që nxirret nga imazhi. Prandaj, për të arritur rezultate sa më të sakta dhe të besueshme, ne duhet të kemi parasysh karakteristikat e imazhit, siç janë nivelit e detajit, ndryshueshmëria, dhe ekuilibri i kontrastit.

Duke ndryshuar përmasat e blloqeve nga 2x2 në 4x4 dhe më pas në 8x8, kemi vërejtur një ndryshim të rëndësishëm në imazhet e masave statistikore. Me madhësi më të vogla blloku, imazhet e vlerave mesatare, devijimi standard, minimale dhe maksimale u shfaqën më të pikseluara, pasi llogaritjet bazoheshin në një numër më të vogël pikselësh. Megjithatë, me rritjen e madhësisë së bllokut, pati një humbje të detajeve dhe imazhet u bënë më të paqarta. Kjo për shkak se masat statistikore u llogaritën në bazë të një numri më të madh pikselësh, duke siguruar një paraqitje më të butë, por më pak të detajuar të imazhit.

Përveç vëzhgimeve të mësipërme, vumë re gjithashtu se me madhësi më të mëdha blloku, imazhi i energjisë bëhej më i vështirë për t'u interpretuar. Me një madhësi blloku 8x8, imazhi i energjisë dukej pothuajse plotësisht i njëtrajtshëm dhe i mungonte ndonjë model i dallueshëm. Kjo mund t'i atribuohet faktit se më shumë blloqe janë përdorur për të llogaritur vlerat e energjisë, duke çuar në një humbje të detajeve dhe një paraqitje më pak të saktë të imazhit.

Përveç kësaj, është e rëndësishme të kemi parasysh në ndryshimin e efekteve të bllokut në atë që ekzekutohet edhe në algoritmit. Nëse blloqet janë shumë të vogla, llogaritjet mund të jenë shumë të ngarkuara për CPU-në, ndërsa nëse blloqet janë shumë të mëdha, bëni këto informacione të informacionit dhe rezultate të tjera të pakta të sakta.

Kështu, ndërsa rritja e madhësisë së bllokut mund të çojë në imazhe më të buta të masave statistikore, mund të rezultojë gjithashtu në humbje të detajeve dhe informacionit të rëndësishëm në raste të caktuara. Është e rëndësishme të merret parasysh imazhi specifik dhe rezultati i dëshiruar kur zgjedhim madhësinë e duhur të bllokut për një aplikacion të caktuar.

Përfundim

Si përfundim, ne kemi eksploruar konceptin e kompresimit të imazhit duke përdorur Transformimin Diskret të Kosinust (DCT) dhe qasjen e bazuar në bllok. Nëpërmjet aplikimit të DCT, ne ishim në gjendje të transformonim të dhënat e imazhit nga domeni hapësinor në domenin e frekuencës, duke rezultuar në një paraqitje më kompakte të imazhit. Duke e ndarë imazhin në blloqe dhe duke kompresuar secilin bllok individualisht, ne ishim në gjendje të arrinim një nivel më të lartë kompresimi duke ruajtur cilësinë e përgjithshme të imazhit.

Për më tepër, ne kemi ekzaminuar ndikimin e madhësisë së bllokut në cilësinë e imazheve të kompresuara. Ne vumë re se madhësitë më të vogla të bllokut rezultuan në një nivel më të lartë detajesh dhe saktësie në masat statistikore, por me koston e një madhësie më të madhe skedari. Nga ana tjetër, madhësitë më të mëdha të bllokut prodhuan një paraqitje më kompakte të imazhit, por në kurriz të një niveli më të ulët detajesh dhe saktësie në masat statistikore.

Në përgjithësi, qasja e bazuar në bllok për kompresimin e imazhit ofron një metodë të fuqishme për reduktimin e kërkesave për ruajtjen e imazheve të mëdha dhe për transmetimin e tyre në mënyrë më efektive përmes rrjeteve. Megjithatë, zgjedhja e madhësisë së bllokut dhe raportit të ngjeshjes duhet të konsiderohet me kujdes për t'u siguruar që imazhet e kompresuara të jenë të një cilësie të mjaftueshme për përdorimin e synuar.