

USHTRIME

Bazat e Procesimit të Imazhit

Ushtrimet nga Bazat e Procesimit të Imazhit ¹

<http://www.prip.tuwien.ac.at/~yll/BPI/>

Yll Haxhimusa
Institute of Computer Aided Automation
Pattern Recognition and Image Procesin Group 183/2
yll@prip.tuwien.ac.at



Universiteti i Prishtinës
Fakulteti Elektroteknikë

Semestri Dimërorë, Dhjetorë 2003

¹Ky projekt u realizu me ndihmën e Brain Gain Programe Projektit të WUS-Austria. Falenderoj stafin e Pattern Recognition and Image Processing Institute-it, veqanarisht Prof. Dr.tech. Dipl.Ing. Walter G. Kropatsch për ndihmesën e bërë gjatë përpunimit të këtij materiali.

Kapitulli i 1

Informata të përgjithshme

Ky kapitull përmbanë informata mbi organizimin e ushtrimeve.

1.1 Vendi i Mbajtjes së Ushtrimeve

Ushtrimet mbahen në laboratorin e kompjuterëve të fakultetit Elektroteknikë.

1.2 Koha e Mbajtjes së Ushtrimeve

Caktohet në orën e parë të ligjëratave si dhe bëhet ndarja në grupet punuese.

1.3 Modusi i Ushtrimeve

Ushtrimet janë të concipuara në dy blloqe të detyrave, në të cilat temat e ligjëratave do të thellohen. Ky dokument përmbanë pjesën e ushtrimeve me kompjuter dhe pjesën e pyetjeve testuese. Në fund të ushtrimeve do të organizohet një test, për të vlerësuar diturinë e fituar.

1.3.1 Detyrat e Ushtrimeve

Ushtrimet përmbajnë pjesën e ushtrimeve me kompjuter dhe pjesën e pyetjeve testuese:

Ushtrimet me kompjuterë

Detyrat me kompjuterë kanë për qëllin të thellojnë pjesë të caktuara të ligjëratave. Detyrat do të punohen duke përdorur MATLAB-in në grupe deri në tre persona. Resultatet e detyrave mund të dërgohen edhe në email në adresën:

y11@prip.tuwien.ac.at

Koha e dorëzimit të rezultateve do të caktohet gjatë ligjëratave.

Pyetjet kontrolluese

Pyetjet kontrolluese janë të pavarura prej ushtrimeve me kompjuterë, dhe kanë po ashtu për qëllim thellimin e pjesëve të caktuara të ligjëratave. Përgjegjja në pyetje behët individualisht në bazë vullnetare. Kjartësimi i pyetjeve të hapura bëhet gjatë ushtrimeve.

1.3.2 Testimi dhe Vlerësimi

Rezultatet e pjesës së ushtrimeve me kompjuterë duhet të dorëzohen me shkrim. Ky dokument duhet të përmbajë zgjedhjen e detyrave dhe përgjegjjet në pyetjet të shtruara. Vlerësimi bëhet duke marrë parasysh përgjegjjen në pyetje gjatë provimit si dhe përfundimit pozitiv të ushtrimeve, edhe atë në raport 60/40.

Vrejtje: E gjithë kjo vlene në rast të organizimit të provimit.

1.3.3 Softueri

Ushtrimet duhet të kryhen duke përdorur MATLAB-in, i cili së bashku me 'Image Processing Toolbox'-in ofron funksionalitet në lëminë e imazh procesimit, po ashtu është fleksibil për të zgjedhur probleme pa pasë nevojë për ndonjë gjuhë programuese me kompajllër. MATLAB dokumentacionin mund ta gjeni në

<http://www.prip.tuwien.ac.at/~y11/BIP/docs.html>.

1.3.4 FAQ dhe Përgjegjjet

- **Është MATLABI Open Source Software?**
JO. Për të interesuarit ekziston mundësia për të blerë një version studentorë me çmim bazik për MATLAB-in prej 100 US-Dollarë, dhe për Image Processing Toolbox rreth 50 US-Dollarë.
- **Kam dëshirë të absolvojë vetëm ligjëratat- është kjo e mundur?**
PO.
- **Nuk mundem të marr pjesë në ushtrime, ka mundësi me u krye ushtrimet edhe më vonë?**
PO. Mirpo në këtë rast konsultimet kryhen vetëm përmes emailit, për arsye se nuk gjendem në Universitetin e Prishtinës. Për këtë duhet të llogartni në vonesa kohore.

1.3.5 Pyetje Tjera ?

Ne çoftëse edhe pas leximit të këtij dokumenti nuk keni mundur të gjeni përgjegjje në pyetjet tuaja atëhere:

- Lexoni FAQ edhe nje herë,
- Pyetni në kohën e ushtrimeve,
- Dërgoni pyetjen tuaja në email: `y11@prip.tuwien.ac.at` ¹

¹Duke ditur se ky dokument nuk është perfekt do të isha mirënjohës për komentet e Juaja rreth këtij dokumenti. Falemenderit.

Kapitulli i 2

Procesimi i Imazhit me MATLAB

Ky kapitull përmbanë informata bazike mbi përdorimin e MATLAB në imazh procesim. Informatat tjera mbi MATLAB-in mund ti gjeni në <http://www.prip.tuwien.ac.at/~yll/BIP>, apo në <http://www.mathworks.com>.

2.1 Urdhërat Kryesorë për Procesim e Imazhit në MATLAB

2.1.1 Thirrja e MATLAB-it

MATLAB-i thirret nën Windows nëse klikoni në ikonën e Matlabit ose në `start -> run -> matlab`, ose nëse së pari startoni windows shell-in dhe më pas shtypni

```
matlab
```

Pas startimit të matlabit shfaqet command promti i MATLAB-it `>> .` Nëse dëshironi të startoni MATLAB-in pa GUI-in (ang. 'Graphical Unit Interface'), për shkak të efijencës, atëherë duhet që gjatë thirrjes së MATLAB-it të shkruhet edhe opcioni `-nojvm`.

2.1.2 Online Help

Me,

```
>> help function_name
```

shfaqet një përshkrim i funksionit `function_name`. Ky urdhër përdoret më së shumti për të përkujtuar numrin e parametrave hyrës dhe dalës të një funksioni. Nëse dëshirojmë një pasqyrë të funksioneve nën MATLAB atëherë shtypet

```
>> helpwin
```

Me urdhërin

```
>> loolfor theme
```

shfaqet një liste e të gjithë urdhërave në të cilët paraqitet mostra kërkuese **theme**.

2.1.3 Leximi i Imazheve

Urdhëri

```
>> I = imread( 'filename' ) ;
```

futë në variablën I përmbajtjen e imazhit 'filename'. Pikëpresja (;) në fund të urdhërit deaktivizon shfaqjen e rezultatit në ekran. Formatet vijuese janë të përkrahura deri tani nga MATLABi:

- JPEG (Joint Photographics Experts Group),
- TIFF (Tagged Image File Format),
- BMP (Microsoft Windows Bitmap),
- PNG (Portable Network Graphics),
- HDF (Hierarchical Data Format),
- PXC (Paintbrush),
- XWD (X Windwo Dump).

Në rast të një imazhi me nuanca gri, variabla I ësthë një $m \times n$ matricë me m rreshta, dhe n kolona. Intenzitetit të një pikseli i qasemi me

```
>> h = I( 20, 100 ) ;
```

Në rast të një kolor imazhi (RGB), matrica ka madhësi $m \times n \times 3$ (tri shtresa, një shtresë për të kuqën (R), një për gjelbërtën (G), dhe një për kaltërtën (B)). MATLAB-i në shumicën e rasteve i ruan të dhënat me *double float* precizitet (klasa **double**, 64-bit). Për imazh procesim, ky representim i të dhënave nuk është ideal. Për të zvogluar sasinë e kujtesës ('memory') të nevojshme për ruajtjen e imazheve, MATLAB-i përkrah dy klasë tjera **uint8** (8-bit) dhe **uint16** (16-bit). Vektorët e të dhënave të këtyre klasëve zënë të tetën gjegjësisht të gjashtëmbëdhjetën pjëse të një **double** vektorit.

Kujdes, për përdorimin e operacioneve aritmetike në mes të dy imazheve të klasës **uint8**, duhet së pari të bëhet transformimi i imazheve në klasën **double** përmes urdhërit **double()**.

2.1.4 Paraqitja e Imazheve

Urdhëri

```
>> imshow(I), colorbar, axis on
```

hap një dritare me imazhin I në të, me një gri skallë dhe me një sistem të koordinative. Sistemi i koordinatave është një *sistem i koordinatave të piksellëve* (r, c). Nëse dëshirojmë të fusim detaje tjera (p.sh. një shenjë) në imazh, duhet së pari të përdoret urdhëri

```
>> hold on
```

që të pengohet ri-vizatimi i figurës. Një shenjë ('marker') futetet në imazh si një plot i thjeshtë, p.sh.

```
>> plot( 100, 50, 'x' ) ;
```

duke pasur kujdes se urdhëri `plot` punon në *koordinatat hapsinore* (x, y) me renditje të kundërt të indekseve. Në shembullin e mësipërm, shenja 'x' do të vendoset në kolonën 100 dhe në rreshtin 50 të imazhit I. Me

```
>> plot( 100, 50, 'rx' ) ;
```

paraqitet një shenjë 'x' e kuqe. Më shumë detaje rreth urdhërit `plot` gjenden në *Online Help*. Një dritare tjetër (p.sh. për paraqitjen e imazhit të dytë) shfaqet me

```
>> figure( 2 )
```

gjegjësisht, me thjeshtë me

```
>> figure
```

2.1.5 Reprezentimi Vektorial i Imazheve

Për shumë probleme është e arsyeshme që imazhet në vend se të reprezentohen me matrica të paraqiten me vektorë. Nëse imazhi I është i paraqitur si matricë atëherë

```
>> Iv = I(:) ;
```

do të transformoje atë në një vektorë, ashtu që kolonat do të stivosohen. Nëse dëshirohet që rreshtat të stivosohen ('staked') duhet përdorur

```
>> It = I'; Iv = It( : ) ;
```

ose,

```
>> reshape( I', length(I( : )), 1 ) ;
```

nëse dëshirojmë të kthejmë imazhin prej representimit vektorial në matricorë, duhet përdorur urdhërin `reshape`.

2.1.6 Konvolucioni, Korrelacioni dhe Detektimi i Tehut

Është dhënë kodi:

```
>> R = [ 1 1; 1 1 ] ;  
>> I2 = conv2( I, R ) ;  
>> imshow(I, [] ) ;
```

Për konvoluimin ('convolution') dy dimensional përmes `conv2`, bërthama ('kernel') e konvolucionit rrotullohet për 180°, që në rastin tonë nuk do të ketë aspak efekt. Nëse dëshirohet të pengohet rrotullimi i maskës R atëhere kjo realizohet me korrelacion ('correlation'):

```
>> I3 = filter( R, I ) ;  
>> imshow( I3, [] ) ;
```

Maska tjera speciale mund të krijohen me urdhërin `fspecial`. Matrica e zbrazët [] në `imshow` pasqyron vlerat e I2 dhe I3 automatikisht në intervalin e vlerave 0-255. Nëse I2 apo I3 dëshirojmë ti ruajmë në njërin prej formateve të suportuara në MATLAB duhet që vlerat i imazhit të skalohen në mënyrë manuale në intervalin e vlerave 0-255:

```
>> O2 = uint8( round( mat2gray( I2 ) * 255 ) ) ;  
>> imwrite( O2, 'filename' ) ;
```

2.1.7 Ekstrahimi i Vlerave të Piksellëve

Urdhëri

```
>> pixval
```

tregon në mënyrë interaktive, gjatë lëvizjes së miut, vlerat e piksellëve. Ky urdhër paraqet po ashtu edhe distancën euklidiane në mes të dy pikëve nëse tasti i majtë i miut mbahet i shtypur. Nëse dëshirojmë që rezultatet ti ruajme në një variabël, përdorim urdhërin

```
>> P = pixval( I ) ;
```

Pas thirrjes së këtij urdhëri, treguesi i miut mbi imazh shëndrohet në një kryq. Për të fituar koordinatat e piksellit të dëshiruar klikohet miu mbi pikën e dëshiruar. Me urdhërin

```
>> improfile
```

llogariten dhe vizualizohen vlerat e intensitetit të piksellëve përgjatë një vijë, gjegjësisht përgjatë një poligoni.

2.2 Këshillëza mbi Progamimin me MATLAB

2.2.1 Gabimet më të Përseritura

- Për të shumëzuar piksellët e dy imazheve përdoret operatori `.*`. Me operatorin `*` MATLAB-i ekzekuton një multiplikim matricorë, që pas një llogaritje të gjatë do të jep një rezultat pa kuptim.
- Në fund të çdo urdhëri do të duhej të shkruhet edhe një pikepresje `;`, përndryshe të gjitha rezultatet do të paraqiten në ekran, që mund të zgjijet shumë.
- Përdormi i `for` cikleve duhet mundësisht të mos përdoren, pasi që programet me `for` cikle kërkojnë kohë të gjatë për të llogaritur.

2.2.2 Vektorizimi i Kodit të MATLAB-it

Indeksimi në MATLAB

Një veti e mirëseardhur e MATLAB-it është aftësia për të zgjedhur një nënbashkësi të elementeve nga një matricë. Në MATLAB ekzistojnë dy mënyra për indeksimit:

Index subscripting - vlerat e indeksëve janë indeksat e vlerave të dëshirueshme të matricës. P.sh. nëse `A = 1:5`, atëherë `A([3, 5])` shfaqë elementin e tretë dhe të pestë të matricës `A`:

```
>> A = 1:5 ;  
>> A([3, 5])  
ans =  
  
3 5
```

Logic Indexing - matrica e indeksave është e madhësisë së njejtë sikurse `A` dhe përmbanë vetëm vlerat 0 apo 1. Elementet e zgjedhura të matricës `A` kanë vlerën '1' në pozicionet gjegjëse të matricës së indeksëve. P.sh. nëse `A = 1:5`, atëherë `A( logical( [0 0 1 0 1] ) )`, shfaqë elementin e tretë dhe të pestë të matricës `A`:

```
>> A = 1:5 ;  
>> A( logical( [0 0 1 0 1] ) )  
ans =  
  
3 5
```

Metoda e dytë e indeksimit është mjetë shumë i fuqishëm. Më shumë informata mbi indeksimin gjenden në *Using MATLAB Manual*.

Operacionet themelore në vektorë: $y(i) = fnc(x1(i), x2(i), \dots)$

Mënyra më e thjeshtë e vektorizimit në MATLAB mund të shiqohet si një *përpunim masiv* i të dhënave. Gjatë kësaj qasje operacioni i njejtë do të ekzekutohet për çdo element të një bashkësie të të dhënave, ku bashkësia e të dhënave përmbanë më shumë se një maticë.

Supozojmë se kemi të dhëna të fituara nga një eksperiment. Të dhënat janë fituar nga matjet e gjatësisë L, gjërsisë B, lartësisë H dhe masës M të një objekti, kërkohet dendësia D e objektit. Nëse eksperimenti do të kryhej vetëm një herë, atëherë do të kishim vetëm nga një vlerë për të katër njësitet matëse (d.m.th L, B, H, dhe M janë skalarë). Llogaritja vijuese do të kryhej për të gjetur dendësinë:

```
>> D = M / ( L * B * H )
```

Nëse eksperimenti është kryer 20 herë, atëherë L, B, H dhe M do të jenë vektorë të madhësisë 20, kërkohet vektori D, që paraqet dendësinë për secilin cikël eksperimental. Në gjuhët programuese do të shkruhej një cikël ('loop'), që në MATLAB do të kodohej si në shembullin vijues, për të kryer llogaritjet:

```
>> for i = 1:20
D(i) = M(i) / ( L(i) * B(i) * H(i) ) ;
end
```

Eleganca e MATLAB-it qëndron në atë, se mund të ignorojmë se eksperimenti është kryer 20 herë. Llogaritja element-për-element mbi një vektor bëhet (po thuasje) më të njetën sintaksë sikurse në shembullin e mësipërm me një element:

```
>> D = M ./ ( L .* B .* H ) ;
```

Dallimi i vetëm qëndron në përdorimin e operatorëve .* dhe ./ . Kjo dallon operatorët element-për-element, *Array Operators*, prej operatorëve të ndryshëm të algjbrës lineare, të ashtu quajtur, *Matrix Operators*.

Operacionet logjike në vektorë: $y(i) = bool(x1(i), x2(i), \dots)$

Një zgjerim logjikë i idesë së *përpunim masiv* është vektorizimi i krahasimeve dhe vendimeve. Le të supozojmë se gjatë kryerjes së eksperimenteve të epërme, na shfaqen vlerat negativë për masën M. Është e qartë se këto vlera janë të gabueshme, andaj eksperimentet gjëgjëse duhet të shpallen të pavlefshme. Eksperimentet e vlefshme mund të gjenden duke aplikuar operatorin >= në vektorin M:

```
>> M = [ -.2 1.0 1.5 3 -1 4.2 3.14 ] ;
>> M >= 0
ans =

0 1 1 1 0 1 1
```

Tani mund të përdoret MATLABI për të larguar vlerat e gabueshme:

```
>> D = D( M >= 0 ) ;
```

Kjo shprehje krijonë një nënbashkësi D, prej elementeve jonegative gjegjëse të M.

Nëse të gjitha elementet janë negative, atëherë duhet të shfaqet një mesazh i gabimit. Ndonjëherë është e nevojshme që vlerat e vektorit logjikë të përmbledhen. MATLAB ofron dy operatorë logjikë në vektorë, **any** dhe **all**, të cilët egzekutojnë operatorët AND dhe OR në tërë vektorin. Në këtë mënyre mund të implementohet testi:

```
>> if (all ( M<0 ) )  
disp( 'Warning: all values of mass are negative.' ) ;  
return ;  
end
```

Po ashtu është e mundur që dy vektorë të madhësisë së njejtë të krahasohen me operatorë të Bool-it

```
>> ( M >= 0 ) & ( L > H )
```

rezultati është një vektor, i madhësisë së njetë sikurse vektori hyrës. Tentimi për të krahasuar dy matrica të madhësive të ndryshme nuk është i lejueshëm.

Konstruimi i vektorëve nga matricat: $y(:,i) = f(x(:))$

Gjenerimi i matricave të thjeshta (p.sh. matricës konstante) është në MATLAB shume thjeshtë. Kodi

```
>> A = ones( 5, 5 ) * 10 ;
```

gjeneron një matricë ku çdo element ka vlerën 10. Shumëzimi në këtë shembull nuk është i nevojshëm, si do të shpjegohet më poshtë. Një problem që paraqitet rregullisht, është gjenerimi i matricës nga elementet e vektorëve. Një matricë, që p.sh. elementet [2, 3; 5, 6; 8, 9; 11, 12; ...] të një vektori x i përgjigjen, gjenerohet në këtë mënyrë:

```
>> x = 1:51 ;  
>> x = reshape( x, 3, length(x) / 3 ) ;  
>> A = x( 2:3, : )' ;
```

Indeksi do të përdoret ashtu që simetria e matricës të shfytëzohet. Që një vektorë i madhësisë $M \times 1$, N herë të duplikohet, përdoret një trik (triku i Tonit), ashtu që të krijohet matrica e madhësisë $M \times N$. Në këtë trik, kolona e parë e vektorit do të indicohet (referohet, gjegjësisht zgjedhet) N herë:

```
>> v = ( 1:5 )' ;
>> M = v( :, ones( 3, 1 ) )
M =
    1  1  1
    2  2  2
    3  3  3
    4  4  4
    5  5  5
```

Në këtë mënyrë mundet të krijohet matrica 5×5 , në të cilën të gjitha elementet kanë vlerën 10:

```
>> A = 10 ;
>> A = A( ones( 5 ) )
A =
    10  10  10  10  10
    10  10  10  10  10
    10  10  10  10  10
    10  10  10  10  10
    10  10  10  10  10
```

Triku i njejtë mund të përdoret edhe në rresht-vektorët, ashtu që indekset ndrojnë renditjen.

Operatorët për bashkësi, për numra dhe sortim

Llogaritjet e bërë deri tani ishin në elementet e vektorëve, edhe nuk vareshin nga elementet tjera të po të njejtit vektorë. Në shumë aplikacione, llogaritjet varen shumë nga vlerat e elementeve tjera, p.sh. nëse punohet më një vektorë x , që paraqet një bashkësi. Për të vendosur nëse një element është redundant dhe duhet të largohet, mund të vendoset vetëm nëse të gjithë elementet tjerë të përpunohen. Në fillim nuk është e kjo se si të largohen këto vlera redundante pa përdorimin e cikleve (loops). FunkSIONET vijuese mund të përdoren për qëllime të tilla:

```
max    elementi maksimal
min    elementi minimal
sort    sortimi i elementeve ne renditje rritese
diff    operatori diferencues. Per nje vektor X, diff(X) eshte:
        [X(2)-X(1) X(3)-X(2) ... X(n)-X(n-1)].
find    gjenden elementet jo te barabart me 0 dhe NaN ('Not a Number')
```

Në lidhje me elementet redundante vlenë kjo veti: Nëse një vektorë është i rradhitur ('sorted'), atëherë elementet redundante janë fqinjë. Po ashtu për dy vlera të njehta operatori `diff` rezulton në vlerë zero. Andaj qëllimi është që të gjenden vlerat e vektorit të sortuem, që kanë diferencë zero :

```
% Sprova e pare. Jo ne menyre korrekte!
>> x = sort( x( : ) ) ;
>> difference = diff( x ) ;
>> y = x( difference ~= 0 ) ;
```

Ky version është deri diku korrekt - duhet pasur parasysh se funksioni `diff` jep si rezultat një vektor dalës që ka një element më pak se vektori hyrës. Në algoritmin e mësipërm elementi i fundit nuk mirret parasysh. Kjo mund të korrigjohet nëse vektorit x para se të përdoret operatori i diferencës i shtohet një element. Ky element duhet patjetër të jetë i ndyshueshëm nga elementi i parafundit, një element i tillë që duhet shtuar është NaN (Not a Number):

```
% Versioni perfundimtar
>> x = sort( x( : ) ) ;
>> difference = diff( [ x; NaN ] ) ;
>> y = x( difference ~= 0 ) ;
```

Dy komente për algoritmin e mësipërm: operatori `(:)` përdoret për të qënë të sigurt se x është vektore, së dyti funksioni `find` përdoret në këtë rast për arsye se ky funksioni nuk jep si rezultat indekset për NaN elementet.

Le të supozojmë se jemi të interesuar për të ditur së sa herë një element paraqitet në matricën origjinale. Pas sortimit të x , përdoret `find` për të gjetur indekset, në pozitat ku shpërndarja ndyshon. Diferenca në mes të dy indekseve të njëpasnjëshme paraqet numrin e paraqitjeve të këtij elementi.

```
% Llogaritja e redundances ne nje vektore x
>> x = sort( x( : ) ) ;
>> difference = diff( [ x; max( x ) + 1 ] ) ;
>> count = diff( find( [ 1; difference ] ) ) ;
>> y = x( find( difference ) ) ;
>> plot( y, count )
```

Kapitulli i 3

Detyrat Laboratorike me Kompjuterë

Çdo grupë (jo secili individualisht) dërgon rezultatet e detyrave në mënyre elektronike. Duhet dorëzuar Matlab fajllat (*.m) si dhe një **PDF dokument**, në të cilin të gjitha detyrat janë zgjedhë së bashku me përgjegjjet në pyetjet e shtruara. **Të gjithë fajllat duhet të komprimohen në një arhivë në format 'platform independent', p.sh. .zip, .tar.gz, tgz**. Emri i arhivës duhet të jetë i njejtë me emrin e grupit, p.sh. për grupin e shtatë

```
bip07_ushtimet.tar.gz  ose  
bip07_ushtimet.tgz    ose  
bip07_ushtimet.zip
```

3.1 Shembulli 1: Njoftimi me MATLAB

Lexoni 'MATLAB primer' që mund ta gjeni në web faqën
<http://www.prip.tuwien.ac.at/~yll/BPI/docs.html>.
Zgjedhni detyrat e mëposhtme si MATLAB skripta ('script'), imazhet gjinden në: <http://www.prip.tuwien.ac.at/~yll/BPI/images.html>.

3.1.1 Vektorët

1. Krijoni një vektorë $\mathbf{t}$ me N komponente në mënyrë lineare dhe rritëse prej 0 deri në 2π , ku $N = 5, 10$. Për $N = 5$ vektori duhet të duket [0.0000 1.5708 3.1415 4.71239 6.28319]
2. Krijoni një vektorë $\mathbf{y}$ të madhësisë së njetë sikurse $\mathbf{t}$, ku $y_i = \sin(t_i), i = 1, \dots, N$. Vizatoni $\mathbf{y}$ në varshmëri të $\mathbf{t}$ për $N = 5, 10, 100$ në subplots-a të ndryshëm.

- Shkruani një MATLAB funksion `gaus1d(x, sigma)`, që për parameter hyrës ka vektorin $\mathbf{x}$ dhe skalarin σ kurse si rezultat jep vektorin $\mathbf{y}$, ku për elementin e x_i -të të vektorit hyrës, llogaritet vlera dalëse y_i si vijonë:

$$y_i = \frac{1}{\sqrt{2\pi}\sigma} \exp -\frac{x_i^2}{2\sigma^2} \quad (3.1)$$

Vizualizoni Eq. 3.1 në intervalin $x \in [-15, 15]$, për $\sigma = 1, 2, 3$ dhe me hap prej 0.1 në një figurë ('plot') me ngjyra të ndryshme. Titulloni boshtet me `xlabel` dhe `ylabel` dhe aktivizoni `grid`-un.

3.1.2 Matricat

- Shkruani një funksion $P(n)$ që krijon një matrice të dhënë si vijon:

1 1 1 1 1 1	1 1 1 1 1 1 1
1 2 2 2 2 1	1 2 2 2 2 2 1
1 2 3 3 2 1	1 2 3 3 3 2 1
1 2 3 3 2 1	1 2 3 4 3 2 1
1 2 2 2 2 1	1 2 3 3 3 2 1
1 1 1 1 1 1	1 2 2 2 2 2 1
	1 1 1 1 1 1 1

- Vizualizoni matricën $P(6)$ në dy `subplots`-a të ndryshëm, njëherë me urdhërin `mesh`, dhe herën tjetër me urdhërin `imshow`.

Këshillëz: Përdorni urdhërin e Matlabit `help` për një përshkrim të shkurtër të urdhrave, p.sh. `help mesh`. Gjatë përdorimit të urdhërit `imshow` është e nevojshme që të bëhet përshtatja e paletës së ngjyrave përmes: `imshow(Image, [])`. Maticat e vogla mundën të vizualizohen më mirë përmes opcionit '`notruesize`'.

- Vizualizoni një funksion Z me $x, y \in [2\pi, -2\pi]$ dhe $\sigma = \pi/2$ në dy `suplots`-a me urdhërin `imshow`:

$$Z = \exp\left(-\frac{X^2 + Y^2}{2\sigma^2}\right) \sin(2X) \quad (3.2)$$

3.1.3 Vlera Mesatare dhe Devijimi Standard

- Krijoni dy sinjale një dimensionale nga imazhi '`lena.tif`' Fig. 3.1.

```
>> Lena = double( imread( 'lena.tif' ) );
>> y1 = Lena( 2, : ) ;
>> y2 = Lena( 200, : ) ;
```



Figura 3.1: Lena

Llogaritni vlerën mesatare ('mean') dhe devijimin standart ('standard deviation' = devijimi mesatar prej vlerës mesatare) për dy sinjalet e mësipërme.

2. Imazhi '`lena.tif`' duhet të ndahet në blloqe të madhësisë 8×8 . Në secilin prej këtyre blloqeve llogaritni vlerën mesatare dhe devijimin standart, më pas ruani këto rezultate në një imazh të madhësisë 32×32 . Vizualizoni imazhet dhe interpretoni rezultatet.

3.2 Shembulli 2: Zhurma dhe Rrafshimi i Imazhit

Zhurma ('noise') në imazh procesim është pjesë (e pandarshme) e krijimit të imazhit, dhe në të shumtën e rasteve modelohet si zhurmë aditive. Për një imazh $m \times n$ kjo modelohet si:

$$f(i, j) = s(i, j) + \eta(i, j), \quad (3.3)$$

ku $f(i, j)$ është vlera e matur e piksellit në pikën në rreshtin i dhe kolonën j , $s(i, j)$ paraqet vlerën e vërtetë të piksellit pa zhurmë, dhe $\eta(i, j)$ është zhurma aditive.

1. Krijoni një 3×3 dhe 7×7 'mean' filtër dhe dy filtra me bërthamë ('kernel') të Gauss-it G_σ me $\sigma = 0.5$ dhe $\sigma = 1.0$. Për krijimin e filtrit mund të përdoret urdhëri `fspecial`. Konvoluoni një gri imazh sipas dëshirës suaj, më të katër filtrat. Përdorni për këtë urdhërin `conv2`. Rënditni rezultatet në mënyrë tabelare dhe diskutoni rezultatet.
2. Shtoni imazhit të mësipërm zhurmë 'salt&pepper' me dendësi ρ . Bëni eksperimente me $\rho = 0.05, 0.1, 0.2$. Për këtë përdorni urdhërin `imnoise` të MATLAB-it. Llogaritni gabimin mesatar quadratic MSQ ('Mean Square Error') në mes imazhit origjinal dhe imazhit me zhurmë si vijon:

$$\text{MSQ} = \frac{1}{mn} \sum_{i=1}^n \sum_{j=1}^m [f(i, j) - s(i, j)]^2 \quad (3.4)$$

Shmanguni përdorimit të `for` cikleve me çdo kusht.

3. Përsëritni pikën nën 1 por me imazh më zhurmë $\rho = 0.05$.
4. Përdorni një 3×3 'median' filtër në imazhin me zhurmë. Për këtë mund të përdoret urdhëri `medfilt2`. Eksperimentoni me kushte kufitare fikse dhe periodike, duke përdorur opcionet '`zeroes`' dhe '`symmetric`'. Sa është MSQ në mes imazhit të filtruar me 'median' filtër dhe atij origjinal? Iteroni filtrimin me 3×3 'median' filtër në imazhin me zhurmë, deri në 10 iterime. Paraqitni vizualisht MQF si funksion të indeksit të iterimeve. Po ashtu vizualizoni, se sa përqind të piksellëve ndyshojnë pas çdo iterimi. Konvergjon rezultati në një imazh fiks? Renditni observimet në mënyre tabelare dhe diskutoni rezultatet.

Këshillëz: Me urdhërin `bar` vizatohet një 'bar' diagram.

3.3 Shembulli 3: Transformimi i Fourier-it

Studioni së pari temën mbi **Transformimet e Fourier-it** si dhe manualët e MATLAB-it dhe 'Image Processing Toolbox'-it (<http://www.prip.tuwien.ac.at/~y11/BPI/>)

3.3.1 DFT dhe Rrafshimi

'Discrete Fourier Transform' paraqet formen diskrete të transformimit të Furies dhe duket vijon:

$$F(k, l) = \frac{1}{n^2} \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} f(i, j) \exp[-i2\pi(\frac{ki + lj}{n})] \quad (3.5)$$

1. Krijoni një Imazh si vijon

```
>> I = zeros( 256, 256 ) ;  
>> I( 138:168, 138:158 ) = 1 ;
```

dhe vizualizoni spektrin e amplitudës (me bazë logaritmike) të DFT së imazhit, në këtë rast përdorni urdhërin `mesh`. Përdorni patjetër po ashtu urdhërin `fftshift`.

2. Rrotulloni imazhin I për 45° dhe paraqitni spektrin logaritmik të amplitudës me një `mesh`. Interpretimi dallimet e domenit të Fourier-it të imazhit të pa rrotulluar nga ai i rrotulluar.

3. Konvoluoni imazhin I me një 3×3 'mean' filtër të thjeshtë, si dhe me një Gauss filtër me bërthamë $G_{0.5}$. Paraqitni spektrin e amplitudave për imazhin origjinal dhe për imazhet të rrafshuara ('smoothed', filtruara), duke përdorur nga një mesh. Përshkruani dallimet të spektrit të amplitudave në mes filtrimit të imazhit me filter të thjeshtë 'mean' dhe Gauss filtër.
4. Filtroni imazhin I me filtër 'mean' me maskë 3×3 , 7×7 dhe 9×9 respektivisht, si dhe me filter të Gauss-it me bërthamë $G_{0.5}$, G_1 dhe $G_{1.5}$, gjegjësisht. Ndani kolonën e 128-të prej matricës së spektrit logaritmik të amplitudës, dhe paraqitni grafikisht, edhe atë për secilin prej filtrave 'mean' dhe për secilin filtër të Gauss-it në një $2 - D$ plot. Kujdes në titullim e boshteve. Përshkruani se cili fenomen forcohet (rritet) me rritjen e maskës së filtrit.
5. Kryeni një rrafshim (lënim) me $G_{1.5}$ jo në domenin hapsirë-kohë por në domenin e Fourier-it. Paraqitni spektrin logaritmik të amplitudës së DFT të imazhit origjinal dhe atij të filtruar me Gauss filtër. Tregoni, se imazhi i transformimit inverz të Fourier-it është identik me imazhin e konvolutuar me bërthamë të Gauss-it.

3.3.2 Restaurimi i Imazhit

1. Imazhi 'noisy.tif' Fig. 3.2 përmbanë zhurmë, i cili është shkaktuar nga një vale sinusoidale me frekuencë dhe drejtim të caktuar. Shprehja $\eta(i, j)$ në Eq. 3.3 është në këtë rast një valë sinusoidale. Duke ditur vetinë e adicionit të DFT-së, imazhi me zhurmë është:

$$F(u, v) + N(u, v) \quad (3.6)$$

ku $F(u, v)$ është DFT e $f(i, j)$ dhe $N(u, v)$ është DFT e $s(i, j)$. Detyra është që kjo inference sinusoidale të largohet nga imazhi.

3.4 Shembulli 4: Operatori i Thjeshtë Tehu

Qëllimi i kësaj detyre është studimi dhe zhvillimi i një operatori të thjeshtë për detektim të tehut ('edge detector'). Përdorni imazhet testuese 'ImageA.jpg' Fig.3.3a dhe 'ImageB.jpg' Fig.3.3b, të cilat mund të gjenden në <http://www.prip.tuwien.ac.at/~y11/BPI/images.html>.

1. Lexoni ImageA.jpg dhe konvoluoni me një 'low-pass' filtër (krijohet me `fspecial`, kujdes në madhësinë korrekte të maskes së filtrit), ashtu që të zvogloni zhurmën.
2. Përdorni funksionin `gradient` për të llogaritur derivatin në drejtimin x dhe y .

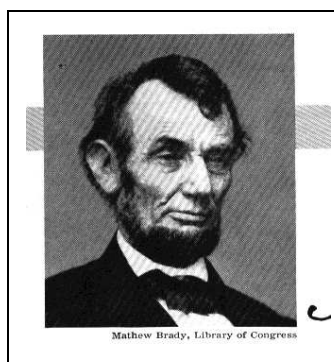


Figura 3.2: Presidenti Lincoln. Vreni zhurmën sinusoidale.

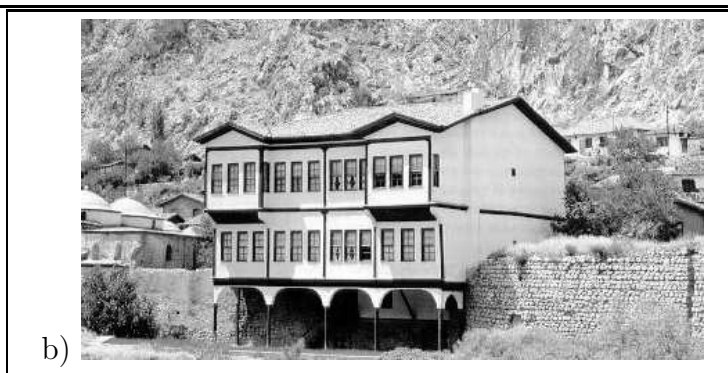
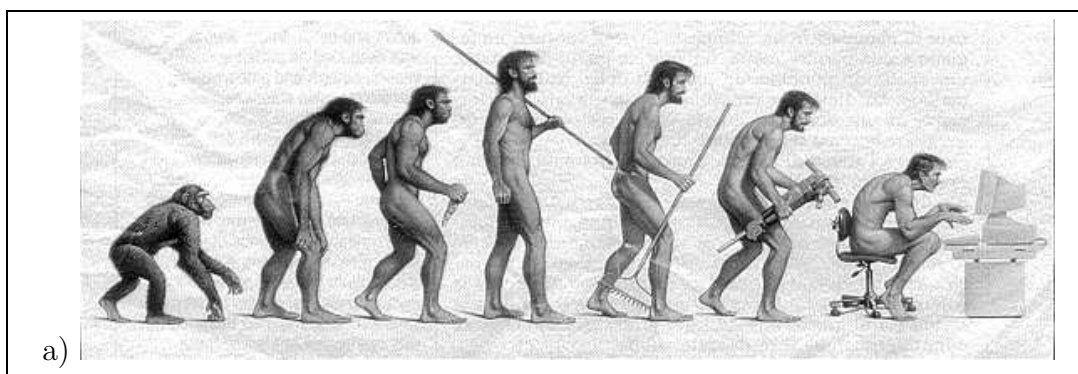


Figura 3.3: Evolucioni dhe Shtëpia 'shqipëtare'.

```
>> [ dx, dy ] = gradient( A ) ;
```

Vizualizoni derivatet me ndihmën e urdhërit `imshow`. Çfarë strukturash mund të shifen në këto imazhe?.

3. Vizualizoni amplitudën e derivatit me ndihmë të `imshow`.
4. Përdorni urdhërin `quiver( dx, dy )` për të paraqitur fushën vektoriale.

Në fushën vektoriale në cdo vend është i definuar një vektor. Paraqini këtë fushë vektoriale së bashku me imazhin gri të amplitudës së invertuar të gradientit. Zoom-oni në imazh dhe interpretoni gjatësinë dhe drejtimin e vektorëve. Vrejtje: për të paraqitur fushën vektoriale së bashku me imazhin gri të amplitudës së gradientit duhet kjo renditje e urdhërave të egzekutohet:

- 1) `imshow`
- 2) `hold on`
- 3) `quiver`

kjo radhitje siguron që `quiver` të përdorë të njejtin sistem koordinativ me `imshow`.

5. Krijoni një imazh binarë `B` përmes një pragu ('threshold') në amplitudën e derivatit `G`:

```
>> B = G > T ;
```

Përdorni një prag `T`, ashtu që në imazh tehet ('edges') të kenë sa me pak ndërprerje.

6. T'holloni ('thining') tehet më pas me procedurën vijuese:
 - (a) Vizitoni secilin piksell të imazhit binarë. Për secilin prej ketyre piksellëve shiqoni në drejtim (dx/b , dy/b) (b është amplituda e gradientit në piksellin e tehut). Pasi që ky fqinj nuk është numer i plotë, thjeshtë rrumbullaksohet në njërin prej 8 fqinjëve (funksioni `round`). Në këtë rast këndi nuk do të jetë edhe më tutje korrekt, neglizhohet pasiqë kemi të bëjmë me një operatorë të thjeshtë tehu.
 - (b) Shiqoni në vlerën e amplitudës të gradientit të piksellit fqinj në këtë drejtim dhe në drejtimin e kundërt.
 - (c) Nëse pikseli i vizituar ka një vlerë të amplitudës së gradientit më të vogël se njëri prej fqinjëve të tij atëherë vlerën e amplitudës së piksellit të vizituar vendose në zero.
7. Përdorni operatorin e zhvilluar mësipërm, dhe krijoni një funksion `my_edge` me këtë prototip:

```
function E = my_edge( I, SIGMA, THRESH )
%
% E = my_edge( I, THRESH )
% INPUTS
```

```

%      I      Grayscale Image, size HxW.
%      SIGMA  Standard deviation of the Gaussian filter;
%              the size of the filter is choosen automatically
%              based on SIGMA.
%      THRESH All gradient magnitudes below THRESH are ignored.
%
% OUPUTS
%      E      Binary edge image, size HxW.

```

Përdorni këtë funksion në imazhin ImageB.jpg Fig.3.3b, dhe paraqitni imazhin origjinal dhe atë të t'holluar.

3.5 Shembulli 5: Operatori i Laplas-it

1. Krijoni një maskë të filtrit të Laplas-it si vijonë:

```

0  1  0
1 -4  1
0  1  0

```

Llogaritni transformimin diskret të Fourier-it (DFT) dhe ilustroni këtë rezultat në një **mesh**. Interpretoni vetitë e operatorit të Laplas-it në domenin frekuencore.

Këshillëz: Për një vizualizim kuptimplotë, duhet së pari të rritet rezolucioni në domenin hapsirorë (p.sh. 256×256 , ashtu që çdo element me indeks më të madh se 3 do të jetë zero. Shiqo ushtrimin 3.1 për referencë.)

2. Krijoni një imazh si vijon:

```

>> I = zeros( 256, 256 ) ;
>> I( 50:200, 50:200 ) = 1 ;

```

dhe rrafshojeni me një bërthamë të Gauss-it $G_{1.5}$. Konvoluoni më pas këtë imazh të rrafshuar me operatorin e epërm të Laplas-it dhe interpretoni rezultatet. Çka paraqesin vlerat pozitive/negative? Ku gjenden tehet?

Këshillëz: Mos harroni që gjatë vizualizimit të skaloni automatikisht imazhet (`imshow( I, [] )`).

3. Theksimi ('sharpning') i imazhit ka për qëllim, që kontrastet lokale në mes të piksellëve fqinjë të i forcojë (rrisë) dhe me këtë tehet ti bëjë më të 'mprehta'. Për të ilustruar, se si realizohet kjo me operatorë të Laplas-it, përdorni imazhin e rrafshuar dhe imazhin pas përdorimit të operatorit të Laplas-it, të cilët ishin krijuar në shembullin e përparmë duke ekstatuar së pari

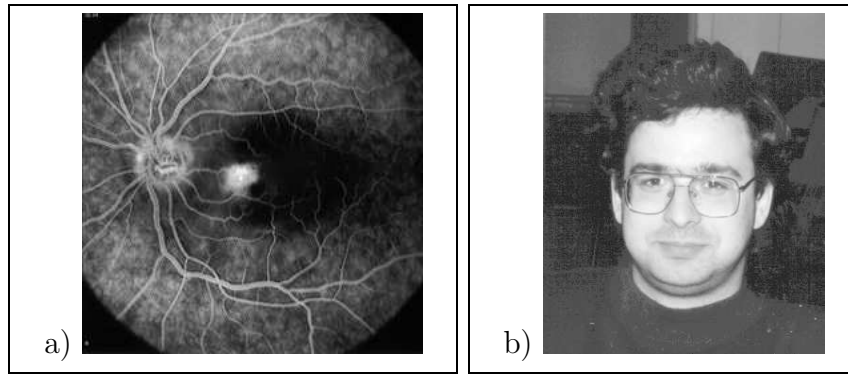


Figura 3.4: Retina e syrit dhe fytyrë njeriu.

një sub matricë (100, 30:70) (kjo jep si rezultat dy vektorë g dhe s të dimensionit 41). Tregoni në një `plot` (me `hold on`) këto informacione:

- g
- s
- $(g-cs)$, për $c = 1, 2$

Përdorni ngjyra të ndryshme dhe/ose tipe të ndryshme të vijave (`help plot`), ashtu që vizatimet të jenë të dallueshme, dhe titulloni këto vija. Interpretoni këtë vizualizim dhe rolin e konstantës c . Përdorni njohuritë e fituara, për të theksuar imazhin 'retina.png' Fig 3.4a dhe 'face.png' Fig 3.4b. Çfarë problemesh shfaqen gjatë theksimit të imazheve për sa i përket zhurmës në imazh?

3.6 Shembulli 6: MATLAB Operatoret e Tehut

1. Përdorni funksionet vijuese në imazhet 'ImageA.jpg' Fig 3.3a dhe 'ImageB.jpg' Fig 3.3a :

- Sobel
- Canny
- LoG (Laplacian of Gaussian)

Përkruani anët pozitive dhe negative të secilit prej operatorëve duke krahasuar njëri tjetrin. Shiqoni me kujdes rezultatet e secilit algoritëm për sa i përket këndeve ('corners'), teheve të lakuara, ruajtjes së zhurmës, etj.

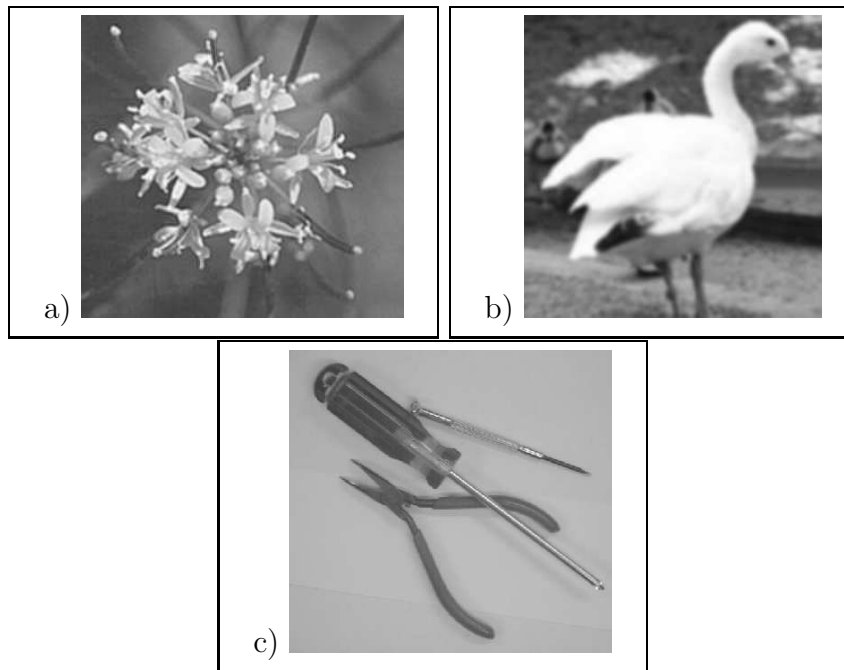


Figura 3.5: a) Lulja, b) mjellma dhe c) mjetet.

3.7 Shembulli 7: Segmentimi përmes Histogrameve

1. Paraqitni histogramet e gri imazheve 'lule.png' Fig. 3.5a, 'mjellme.png' Fig. 3.5b dhe 'tools.png' Fig. 3.5c. Komentoni se çfarë informacionesh mund të nxjerren nga histogramet. Zgjedhni nga një prag ('threshold') për secilin imazh dhe tregoni imazhet binare të fituara.

Këshillëz: Një histogram i një imazhi mundet në këtë mënyrë të krijohet:

```
>> [ N, X ] = histogram( I, 256 ) ;
>> bar( X, N ) ;
```

3.8 Shembulli 8: Hapja/Mbyllja Morfologjike

1. Ekstahoni një imazh binarë nga 'lines.gif' Fig 3.6a përmes hapjes morfologjike me funksionin e Matlab-it `imopen` njëherë në drejtim vertikal e më pas në atë horizontal, d.m.th. krijoni dy imazhe, edhe atë një imazh vetëm me vija vertikale nga 'lines.gif' dhe tjetrin vetëm me vija horizontale.

Këshillëz: Për ekstahimin e vijave vertikale mundet të përdoret p.sh. një element strukturorë i madhësisë 9×3 . Ky element strukturorë krijohet me urdhërin `strel`

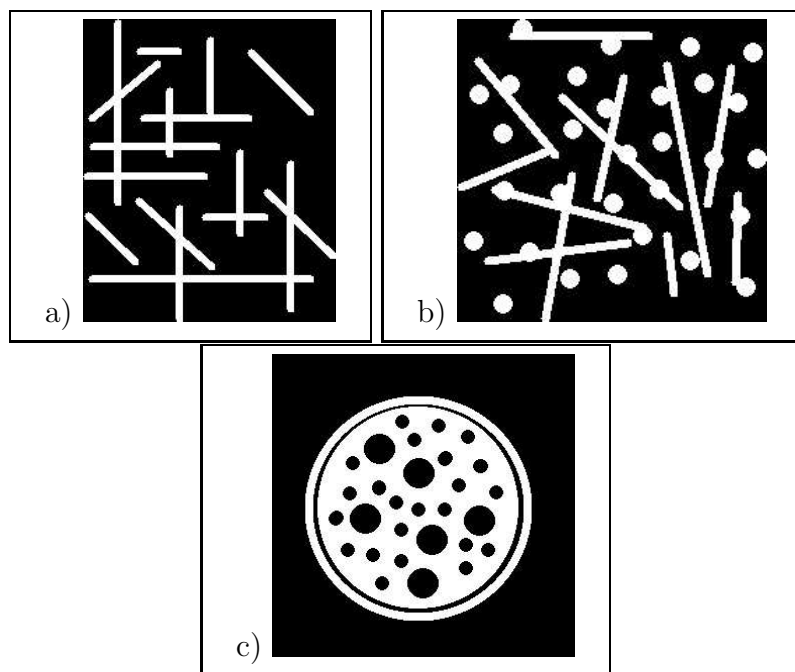


Figura 3.6: a), b), c) Vijat dhe vrimat.

Informata me të hollësishme rreth kësaj teme mund të gjenden në manualin e 'Image Processing Toolbox'-it nën temën 'Mathematic Morphology', shiqo (<http://www.prip.tuwien.ac.at/~y11/BPI/docs.html>)

2. Largoni nga imazhi '`art.gif`' Fig 3.6b vijat duke përdorur hapjen morfologjike. Tregoni se cilin element strukturore keni përdorur për të fituar rezultatin.
3. Largoni nga imazhi '`holes.gif`' Fig 3.6c vrimat e vogla përmes mbylljes morfologjike me urdhërin `imclose`, por duke i ruajtur vrimat e mëdha. Tregoni se cilin element strukturore keni përdorur.

3.9 Shembulli 9: Numrimi i Objekteve

1. Caktoni numrin e fijeve ('fibres') në formë rrethore nga imazhi '`fibres.tif`' Fig 3.7, i cili paraqet prerjen e tërthortë të nje materialin fijorë. Bazë për numrimin e objekteve mund të përdoret p.sh. funksioni `bwlabel`, i cili çdo objekti kompakt në imazh i cakton një numër unik, dhe krijon një të ashtu quajtur etiketë imazh ('label image'). Duke përdorur funksionin `label2rgb` secilës etiketë caktoni një pseudo ngjyrë dhe vizualizoni këtë imazh. Me cilin funksion mundemi nga nje etiketë imazh të i gjejmë numrin e objekteve te

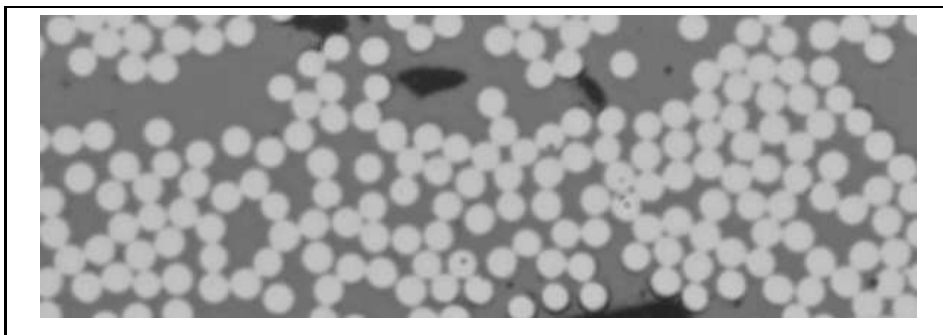


Figura 3.7: Fijet.

kërkuara? Shpjego rezultatin përfundimtar dhe paraqitni edhe rezultatet intermediate.

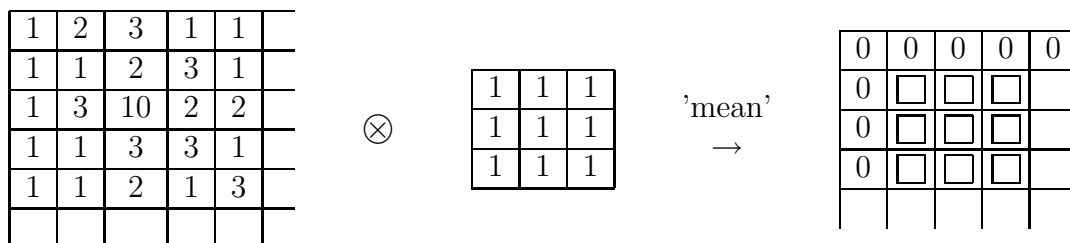
Këshillëz: Objektet në kornizë mundën të mos mirren gjatë procesit të numrimit.

Kapitulli i 4

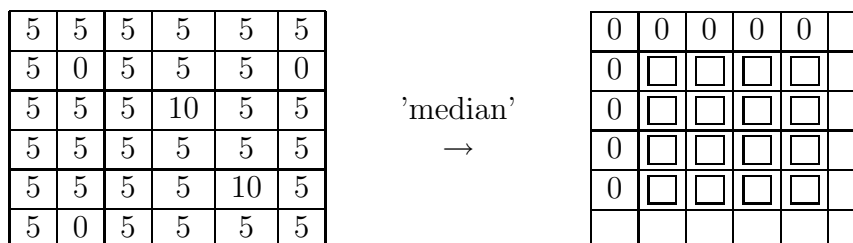
Pyetjet Kontrolluese

Ky kapitull përmbanë pyetjet kontrolluese, me të cilat testohet dituria e fituar gjatë ushtimeve. Përgjegjja është në bazë vullnetare dhe nuk është pjesë e detyrueshme e ushtimeve laboratorike.

1. Konvoluoni imazhin e dhënë me 'mean' filtrin. Llogaritni vlerat e piksellëve në kutitë e theksuar. Me çfarë faktori duhet më pas të normalizohet rezultati.



2. Përdorni një 'median' filtër në imazhin e mëposhtëm edhe llogaritni vlerat e piksellëve në kornizën e theksuar.



3. Është dhënë një imazh I , së pari është rrafshuar (filtruar) me një $n \times n$ 'mean' filtër dhe më pas me një funksion të Gauss-it G_σ . Për të zvogluar kohën e llogaritjeve do të tentohet që së pari 'mean' filtrin të e rrafshojmë (filtrojmë) me funksionin e Gauss-it G_σ dhe kjo maskë rezultuese më pas të përdoret për rrafshimin e imazhit I . Është kjo qasje e drejtë? Shpjegoni përgjegjjen tuaj.
4. Për rrafshimin e një imazhi duhet një $n \times n$ bërthamë $G_{0.5}$ e Gauss-it të përdoret. Sa duhet të jetë i madhë n ?. Emërtoni anët pozitive dhe negative për n shumë të vogël apo shumë të madhë.
5. Shpjegoni se si përdoret Transformi i Fourier-it për konvolumin e një imazhit. Në cilat raste ka kuptim të përdoret DFT.
6. Përse përdoret urdhëri `fftshift` gjatë transformit të Fourier-it ?
7. Emërtoni një aplikacion ku 'high pass' filtrimi përdoret në imazh procesim.
8. Shiqoni imazhin $1D$ e mëposhtëm. Paraqitni në mënyrë kualitative rezultatin e përdorimit të operatorit të derivatit të parë dhe të dytë në intervalin $(p1, p2)$ (vizatoni dy grafiqe për tu përgjegjur pyetjes). Tregoni se ku dhe si mund të lokalizohen tehet ('edge') në intervalin $(p1, p2)$ duke u bazuar në daljen e operatoreve të mësipërm.
9. Është dhënë gri profili (vlerat gri përgjatë një prerje të imazhit) vijues:

2	2	4	2	3	8	11	14	12	13	15	4	2	3	2
---	---	---	---	---	---	----	----	----	----	----	---	---	---	---

Jepni një maske A një dimensionale ashtu që në profin e dhënë të detektohen tehet:

A :

Shkruani më pas gri profilin e tehut. Vendosin vlerat kufitare në zero.

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

10. Shiqoni gri imazhin vijues

1	3	10
2	4	11
3	5	12

Llogartini gradientin dhe amplitudën e gradientit për piksellin në mes, nën përdorimin e operatorit të Sobel-it:

$$\frac{\partial I}{\partial x} \approx \begin{array}{|c|c|c|} \hline -1 & 0 & 1 \\ \hline -2 & 0 & 2 \\ \hline -1 & 0 & 1 \\ \hline \end{array}$$

dhe,

$$\frac{\partial I}{\partial y} \approx \begin{array}{|c|c|c|} \hline -1 & -2 & -1 \\ \hline 0 & 0 & 0 \\ \hline 1 & 2 & 1 \\ \hline \end{array}$$

11. Shpjegoni se pse një operatorë që bazohet në derivatin e dytë ($\partial^2 I / \partial x^2$), mund të përdoret për detektimin e teheve.
12. Pershkruani se çka llogaritë operatori i Laplac-it. Shpjegoni efektin e konvolumit të operatorit të Laplac-it në vendin ku dy regjione homogjene me gri vlera të ndyshme takohen.
13. Shpjegoni se pse Marr-Hildeth (LoG, Laplacian of Gaussian) operatori është një operatorë më i mire se ai i Laplac-it.
14. Çka paraqet 'non-maxima suppression' dhe për çka përdoret në operatorin e Canny-it.
15. Çfare domethënie ka 'threshold hysteresis' dhe përse përdoret në operatorin e Canny-it.
16. Çka është algoritmi për t'hollim të teheve? Ilustroni shkurt një algoritëm për t'hollim.
17. Pergjegjuni në këto pyetje me po (P) ose jo (J):
 - Drejtimi i gradientit është drejtimi në të cilën ndron intenziteti më së shumti.
 - Nuk është e nevojshme të bëhet një 'non-maxima suppression' me LoG operatorin.
 - Amplituda e gradientit mund të llogaritet me konvolucion.
 - Përdorimi i operatorit të Sobel-it me intenzitet konstant c gjeneron vlerën c në tërë imazhin.
18. Çka është përparësia e përdorimit të një maske të madhe gjatë përdorimit të detektorëve të tehut.
19. Çka është përparësia e përdorimit të një maske të vogël gjatë përdorimit të detektorëve të tehut.

20. Shpjegoni hapjen/mbylljen morfologjike në raport me dilatacionin/erosionin.
21. Skiconi rezultatin e dilatacionit të një rrethi, me radius r , me një element strukturorë rrethorë, me radius $\pi/4$.
22. Është dhënë një imazh binarë, që përmbanë një tekst të skenuar anglisht, ku fjalite janë horizontalisht të orjentuara dhe të gjitha shkronjat janë të të njetit font. Skiconi se si me ndihmën e morfologjisë matematike mund të numrohet vokalet që paraqiten në tekst.
23. Çka është teorema e konvolucionit? Çfarë implikimesh praktike ka kjo teoreme në imazh procesim?
24. Në cilat situacione është kuptimplotë të përdoren teknikat e rrafshimit? Krahasoni filtrimi me 'median' dhe 'mean' filtër. Cilat janë anët pozitive/negative të secilës metodë?