

**Digitalna obradba i analiza slike**  
**Izvještaj prve laboratorijske vježbe**

Tomislav Pozaić

Zagreb, 12. ožujak 2012.

### Zadatak 3.1.2

1. *imagesc* automatski skalira podatke i prikazuje sliku. *imshow* je Matlab naredba iz Image Processing Toolboxa koja matricu podataka predstavlja kao sliku. Pretpostavlja da su elementi intenziteti piksela. Za razliku od nje *Image* naredba vizualizira matricu kao matricu brojeva. Brojevi koji predstavljaju boje u tom slučaju nemaju utjecaja na predstavljanje matrice kao slike. Da bi rezultati naredbi bili jednaki potrebno je vrijednosti skalirati. Za naredbu *Image* vrijednosti moraju biti unutar intervala [0,255], a za naredbu *imshow* unutar intervala [0,1].

$$scaled1_i = \frac{255}{\max(imgU(:)) - \min(imgU(:))} * (imgU_i - \min(imgU(:))) \quad //Image$$

$$scaled2_i = \frac{1}{\max(imgU(:)) - \min(imgU(:))} * (imgU_i - \min(imgU(:))) \quad //Imshow$$



**Slika 1.** Image i Imshow

3. U slučajevima kada je dinamički opseg slike tako velik da je samo nekoliko točaka vidljivo, operacijom logaritmiranja može se postići kompresija dinamike sivih tonova. Log funkcija ističe točke male vrijednosti relativno u odnosu na točke velike vrijednosti. Korjenovanjem se postiže slabiji, a kvadriranjem suprotan efekat (Slika 3).

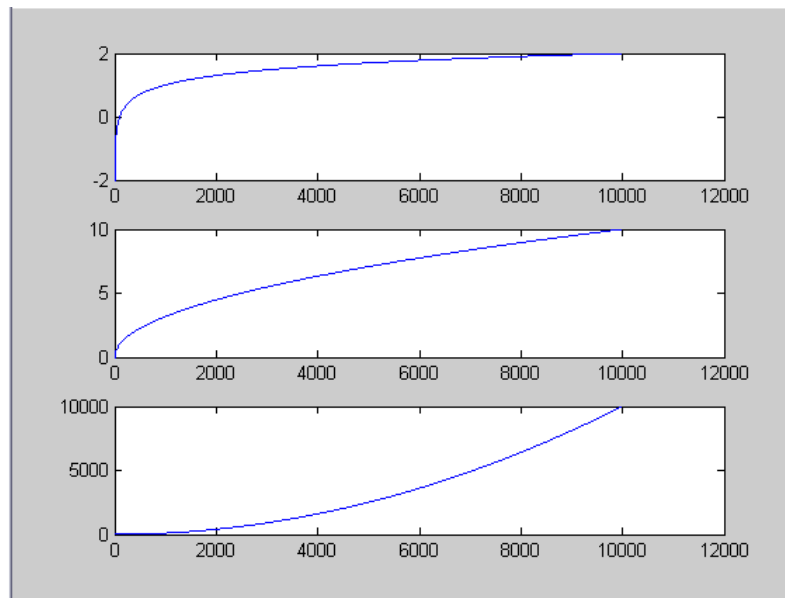
4.

$$H_1 = N \circ U_1 = N(\log(x))$$

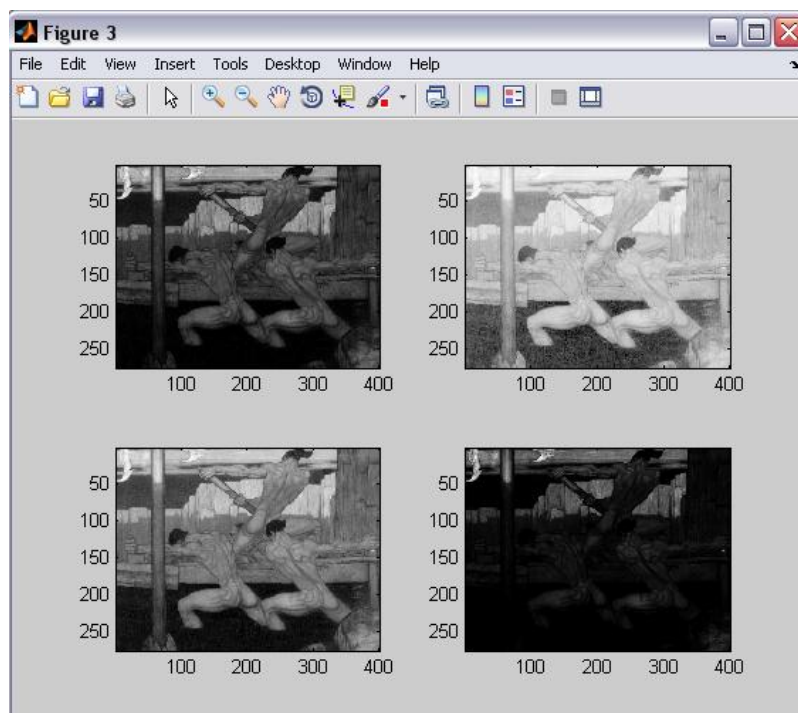
$$H_2 = N \circ U_2 = N(\sqrt{x})$$

$$H_3 = N \circ U_3 = N(x^2)$$

5.



**Slika 2.** Prijenosne funkcije (logaritam, korjen, kvadrat)



**Slika 3.** Prikaz redom originalne slike te slika nakon logaritmiranja, korjenovanja i kvadriranja

```
close all
clear all

cd slike_za_vjezbe
[img,map]=imread('medalja_kamenita_vrata.png');
cd ..
```

```

img=double(img);
imgU=sqrt(img);
scaled1=255*imgU/(max(imgU(:))-min(imgU(:)))-255*min(imgU(:))/(max(imgU(:))-
min(imgU(:)));
scaled2=imgU/(max(imgU(:))-min(imgU(:)))-min(imgU(:))/(max(imgU(:))-
min(imgU(:)));
figure
subplot(1,2,1)
image(scaled1);
colormap(gray)
axis off
axis image
subplot(1,2,2)
imshow(scaled2)
colormap(gray)
axis off
axis image

cd slike_za_vjezbe
[img,map]=imread('uskoci1.png');
cd ..

img = double(rgb2gray(img));

figure
subplot(2,2,1)
imagesc(img);
colormap(gray)

logaritam=log10(img);
logaritam=imgnorm(logaritam);
subplot(2,2,2)
imagesc(logaritam);
colormap(gray)

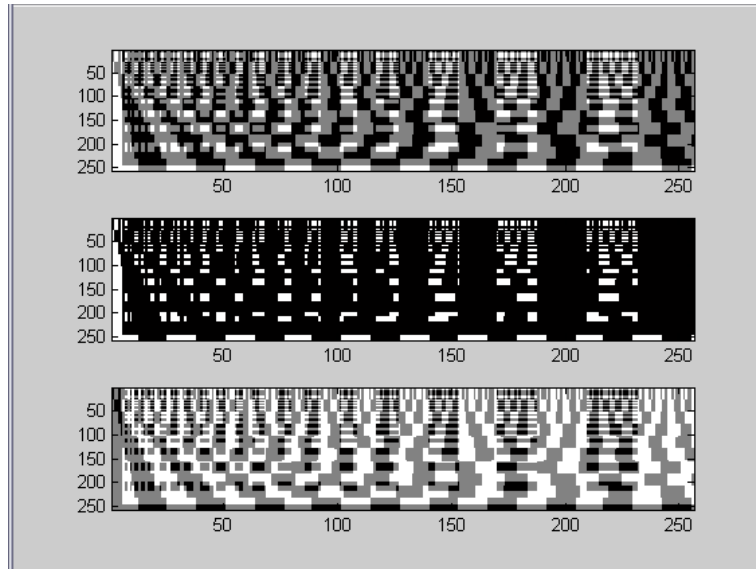
koren=sqrt(img);
koren=imgnorm(koren);
subplot(2,2,3)
imagesc(koren);
colormap(gray)

kvadrat=img.^2;
kvadrat=imgnorm(kvadrat);
subplot(2,2,4)
imagesc(kvadrat);
colormap(gray)

```

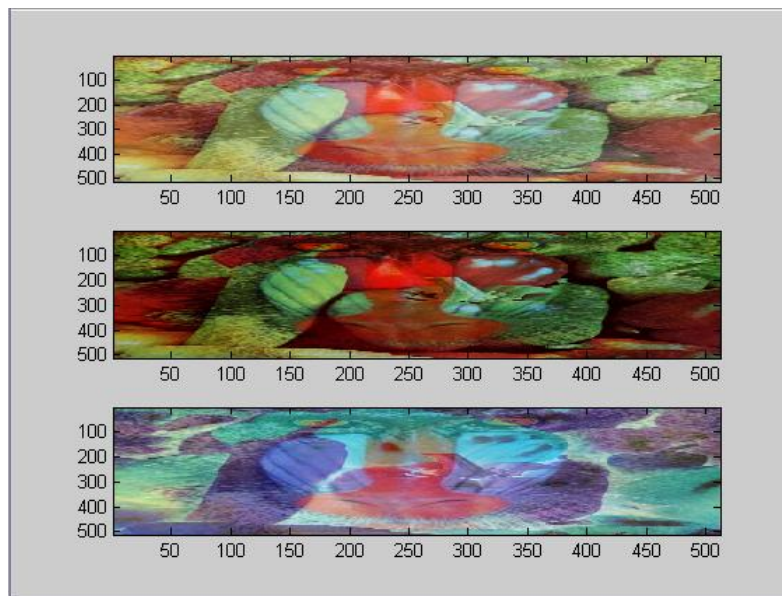
### Zadatak 3.2.2

1. Binarne operacije mogu se koristiti za poboljšavanje slike. Operacije se primjenjuju na individualne piksele tih slika. Veličine slika moraju biti prilagođene (smanjiti veću ili povećati manju sliku). Prostor boja koji obje slike koriste mora biti jednak. Oduzimanje slika, na primjer, može služiti za detektiranje razlika između slika.



**Slika 4.** Dvije slike nakon zbrajanja, množenja i oduzimanja

2.



**Slika 5.** Dvije slike nakon zbrajanja, množenja i oduzimanja

```
clear all
close all

cd slike_zavjezbe
[img1, map] = imread('testpat2.tif');
[img2, map] = imread('uzorak.tif');
cd ..

img2=imresize(img2,[size(img1)], 'bilinear');

zbrajanje=imgnorm(double(img1)+double(img2));
```

```

mnozenje=imgnorm(double(img1).*double(img2));

oduzimanje=imgnorm(double(img2)-double(img1));
figure
subplot(3,1,1)
imagesc(zbrajanje);
colormap(gray)
subplot(3,1,2)
imagesc(mnozenje);
colormap(gray)
subplot(3,1,3)
imagesc(oduzimanje);
colormap(gray)

cd slike_za_vjezbe
[img1, map] = imread('peppers.jpg');
[img2, map] = imread('im9.jpg');
cd ..

img2=imresize(img2,[size(img1,1) size(img1,2)], 'bilinear');

zbrajanje=imgnorm(double(img1)+double(img2));

mnozenje=imgnorm(double(img1).*double(img2));

oduzimanje=imgnorm(double(img2)-double(img1));
figure
subplot(3,1,1)
imagesc(zbrajanje);

subplot(3,1,2)
imagesc(mnozenje);

subplot(3,1,3)
imagesc(oduzimanje);

```

### Zadatak 3.3.1

1. Digitalna supstrakcijska angiografija (DSA) je primjer poboljšanja slike oduzimanjem. Rabi se kod kardiovaskularnih snimanja. Slika pacijenta se snimi prije nego se što se u arteriju ubrizga kontrast koji poboljšava vidljivost arterija u rendgenskoj snimci i ta slika se koristi kao predložak. Ubrizga se kontrast i snimaju se nove slike. Predložak se oduzme od svake snimljene slike tako da samo arterija ostane vidljiva na slici (Slika 6).

```

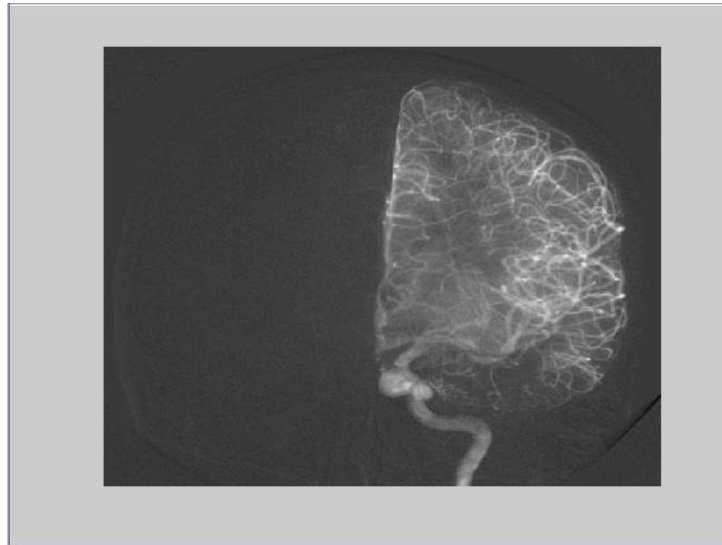
clear all
close all

cd slike_za_vjezbe
[img1, map] = imread('angio0.tif');
[img2, map] = imread('angio1.tif');
cd ..

rez=imgnorm(double(img1)-double(img2));

```

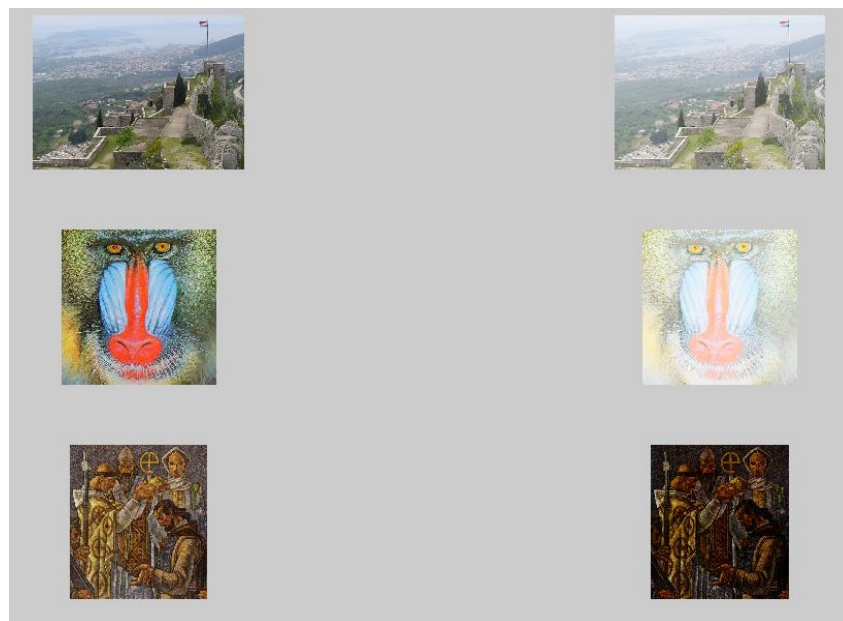
```
figure
imagesc(rez);
colormap(gray)
```



**Slika 6.** Digitalna supstrakcijska angiografija

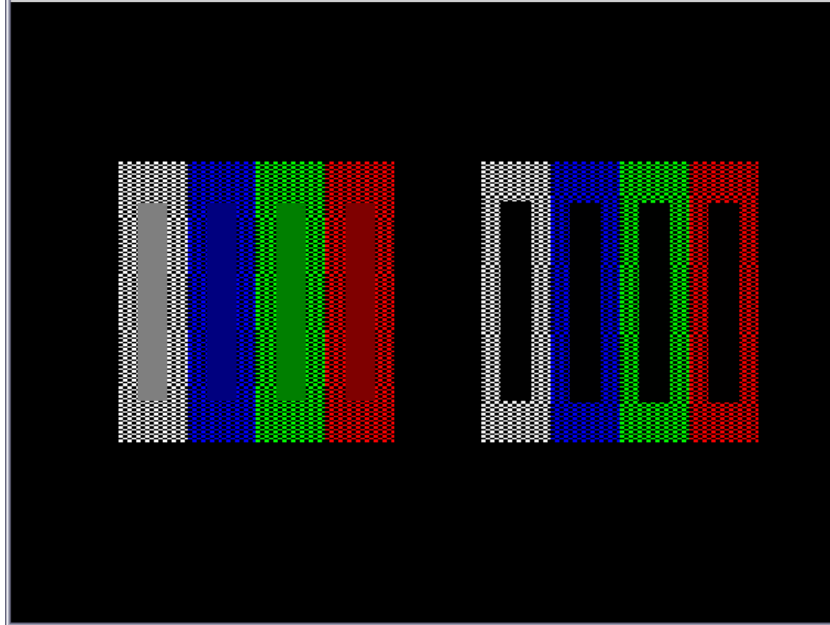
### Zadatak 3.4.2.

1.



**Slika 7.** Originalna slika (lijevo) i slika nakon gama korekcije (0.5, 0.2, 2)

2. Iznosi gama korekcije za koje s veće udaljenosti nije moguće razlikovati popunjeni pravokutnik od odgovarajuće pozadine (crne) iznose [7, 7, 7] (iako već i za vrijednosti veće od [3, 3, 3] s većih udaljenosti se teško razlikuje popunjeni pravokutnik od pozadine).
3. Gama korekcija je unarna operacija na slici (potrebna je samo jedna ulazna slika).



**Slika 8.** Popunjeni pravokutnik u sredini stopio se s pozadinom (lijevo se nalazi originalna slika)

```
clear all
close all

cd slike_za_vjezbe
[img1,map]=imread('split.jpg');
[img2,map]=imread('im9.jpg');
[img3,map]=imread('kljakovic1.png');
cd ..

figure
subplot(3,2,1)
imshow(img1);
imG=imadjust(img1,[],[],0.5);
subplot(3,2,2)
imshow(imG);

subplot(3,2,3)
imshow(img2);
imG=imadjust(img2,[],[],0.2);
subplot(3,2,4)
imshow(imG);

subplot(3,2,5)
imshow(img3);
imG=imadjust(img3,[],[],2);
subplot(3,2,6)
imshow(imG);

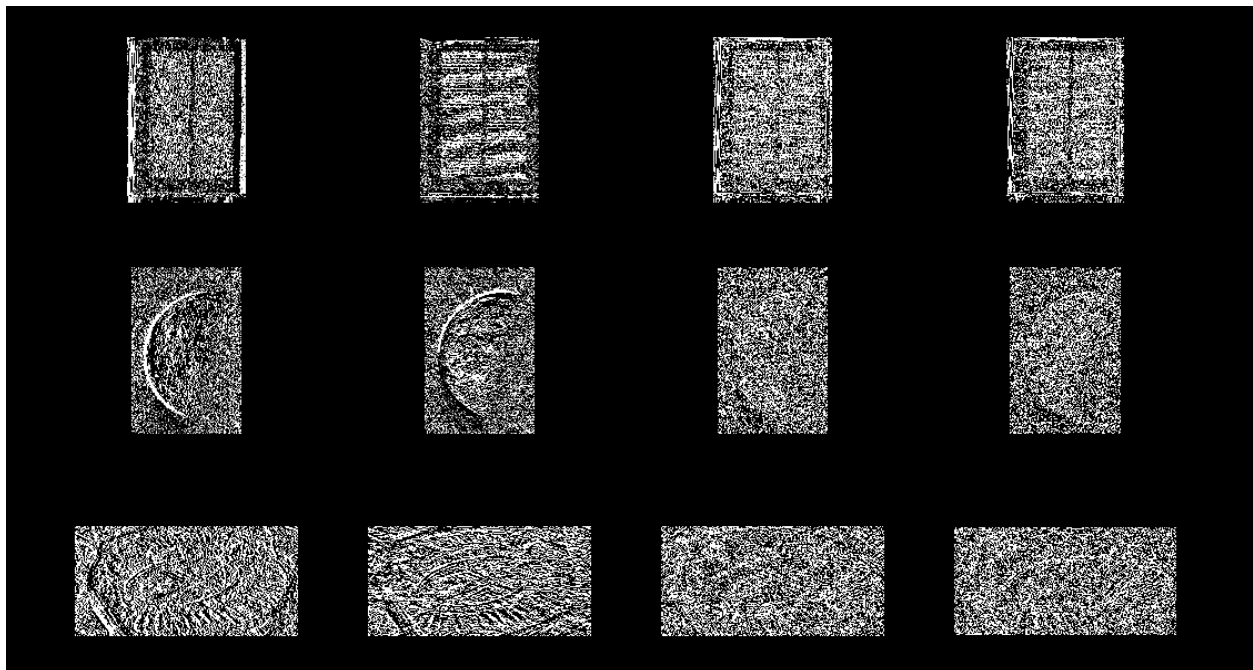
cd slike_za_vjezbe
[img,map]=imread('psgamma.tif');
cd ..
img = ind2rgb(img, map);
figure
```



```
subplot(1,2,1)
imshow(img)
subplot(1,2,2)
imshow(imadjust(img, [], [], [7, 7, 7]));
set(gcf, 'color', [0 0 0]);
```

### Zadatak 3.5.3.

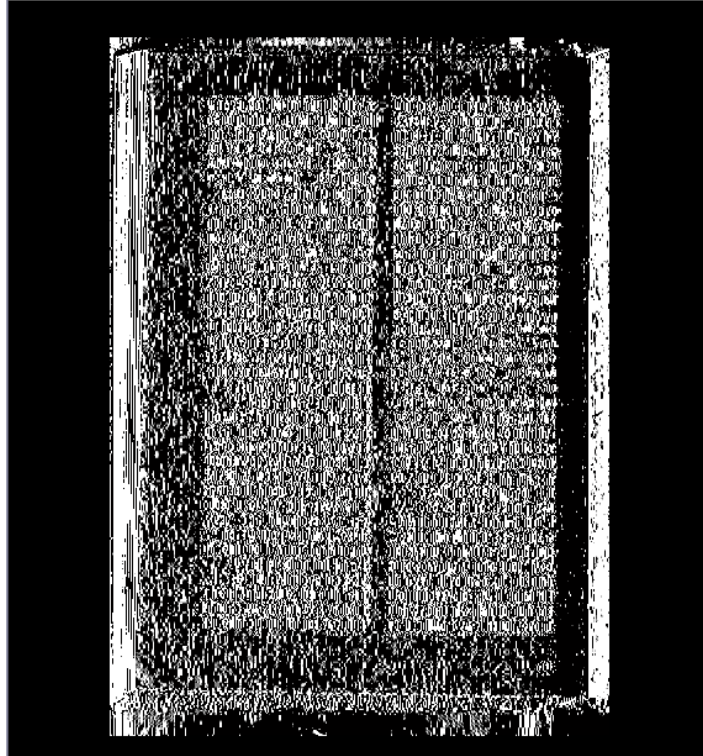
- 1., 2. Prva maska je filter za detekciju rubova u x-smjeru s uteženjem na srednjem pikselu. Druga maska je filter za detekciju rubova u y-smjeru bez uteženja. Ove maske se ustvari koriste za računanje prve derivacije slike. Za računanje druge derivacije slike koriste se tzv. Laplaceovi filtri, a služe također za detekciju rubova. Druga derivacija omogućuje odrediti da li je susjedni piksel rub ili nastavak slike. Treća i četvrta maska su primjeri Laplaceovih filtara.



**Slika 9.** Linearne konvolucije s maskama

- 3., 4. Filtri za usrednjavanje zamjenjuju svaki piksel sa srednjom vrijednošću piksela u kvadratnom prozoru koji okružuje taj piksel. Napravljen je kompromis između uklanjanja šuma i očuvanja detalja. Veći kvadratni prozor može efektivnije ukloniti šum, ali dodatno zamagli detalje/rubove. Maska za usrednjavanje dimenzija 4x4:

$$\frac{1}{16} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$



**Slika 10.** Linearne konvolucije slike "misal\_1483.png" s maskom za usrednjavanje dimenzija 4x4

5. Nije potrebna normalizacija svjetline jer je frekvencijski odziv filtra periodička funkcija koja je u rasponu  $[-0.5, 0.5]$ .
6. Linearna konvolucija je binarna operacija (potrebne su dvije slike odnosno slika i maska da bi se konvolucija mogla realizirati).

```
close all
clear all

cd slike_za_vjezbe
[img1,map]=imread('misal_1483.png');
[img2,map]=imread('moon.tif');
[img3,map]=imread('salona.png');
cd ..

maska1=[1 0 -1; 2 0 -2; 1 0 -1]/4;
maska2=[1 1 1; 0 0 -0; -1 -1 -1]/3;
maska3=[-1 -1 -1; -1 8 -1; -1 -1 -1];
maska4=[0 -1 0; -1 4 -1; 0 -1 0];

imgC1=conv2(img1,maska1,'valid');
imgC2=conv2(img1,maska2,'valid');
imgC3=conv2(img1,maska3,'valid');
imgC4=conv2(img1,maska4,'valid');
imgC5=conv2(img2,maska1,'valid');
imgC6=conv2(img2,maska2,'valid');
```

```

imgC7=conv2(img2,maska3,'valid');
imgC8=conv2(img2,maska4,'valid');
imgC9=conv2(img3,maska1,'valid');
imgC10=conv2(img3,maska2,'valid');
imgC11=conv2(img3,maska3,'valid');
imgC12=conv2(img3,maska4,'valid');

figure
subplot(3,4,1)
imshow(imgC1);
set(gcf,'color',[0 0 0]);
subplot(3,4,2)
imshow(imgC2);
set(gcf,'color',[0 0 0]);
subplot(3,4,3)
imshow(imgC3);
set(gcf,'color',[0 0 0]);
subplot(3,4,4)
imshow(imgC4);
set(gcf,'color',[0 0 0]);

subplot(3,4,5)
imshow(imgC5);
set(gcf,'color',[0 0 0]);
subplot(3,4,6)
imshow(imgC6);
set(gcf,'color',[0 0 0]);
subplot(3,4,7)
imshow(imgC7);
set(gcf,'color',[0 0 0]);
subplot(3,4,8)
imshow(imgC8);
set(gcf,'color',[0 0 0]);

subplot(3,4,9)
imshow(imgC9);
set(gcf,'color',[0 0 0]);
subplot(3,4,10)
imshow(imgC10);
set(gcf,'color',[0 0 0]);
subplot(3,4,11)
imshow(imgC11);
set(gcf,'color',[0 0 0]);
subplot(3,4,12)
imshow(imgC12);
set(gcf,'color',[0 0 0]);

maska=[1 1 1 1; 1 1 1 1; 1 1 1 1; 1 1 1 1]/16;
imgC1=conv2(img1,maska1,'valid');
figure
imshow(imgC1);
set(gcf,'color',[0 0 0]);

```

**Funkcija koja je korištena za skaliranje slika:**

```
function [imgout] = imgnorm(img)
%skaliranje vrijednosti na interval [0,255]
low = min(img(:));
high = max(img(:));
range = high - low;
imgout = 255*(img - low) / range;
end
```