

Poročilo 5. vaje pri predmetu OVS Sivinske preslikave slik

Tina Zwitter 64200432

10. november 2020

1 Minimalne in maksimalne sivinske vrednosti

vrsta preslikave	V bazi int16		V bazi uint8	
vrednost	min vrednost	maks. vrednost	min vrednost	maks. vrednost
originalna slika	-10128	455	0	255
linearna preslikava	199.1250	1522	199	255
linearno oknjenje	0	255	0	255
odsekoma linearna	2.4437	254.0225	2	254
gamma preslikava	0	255	0	255

Tabela 1: Tabela z minimalnimi in maksimalnimi vrednostmi

2 Upragovanje

```
function oImage = thresholdImage(iImage, iT)
%trehsholdImage naredi upragovanje slike.
%vhodni parametri:
% iImage - slika podana v matrični obliki
% iT - parameter upragovanja
%izhodni paramatri:
% oImage - slika pridobljena z upragovanjem v matrični obliki.

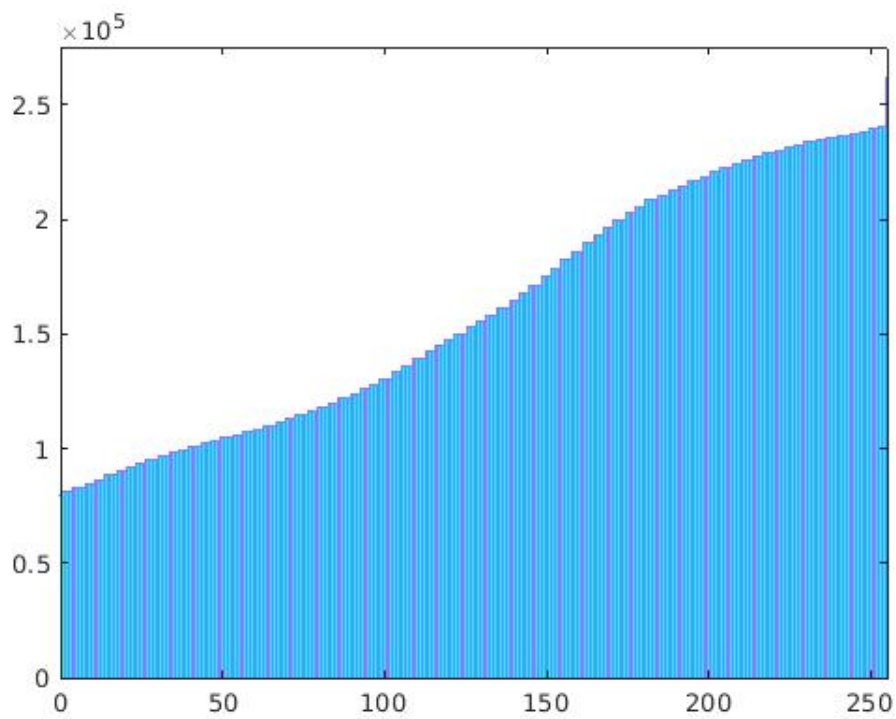
Lg = 2^8 -1;
oImage = iImage;
oImage(iImage<=iT) = 0;
oImage(iImage>iT) = Lg;
end
```

3 Štetje sivinskih vrednosti $s_g = 0$

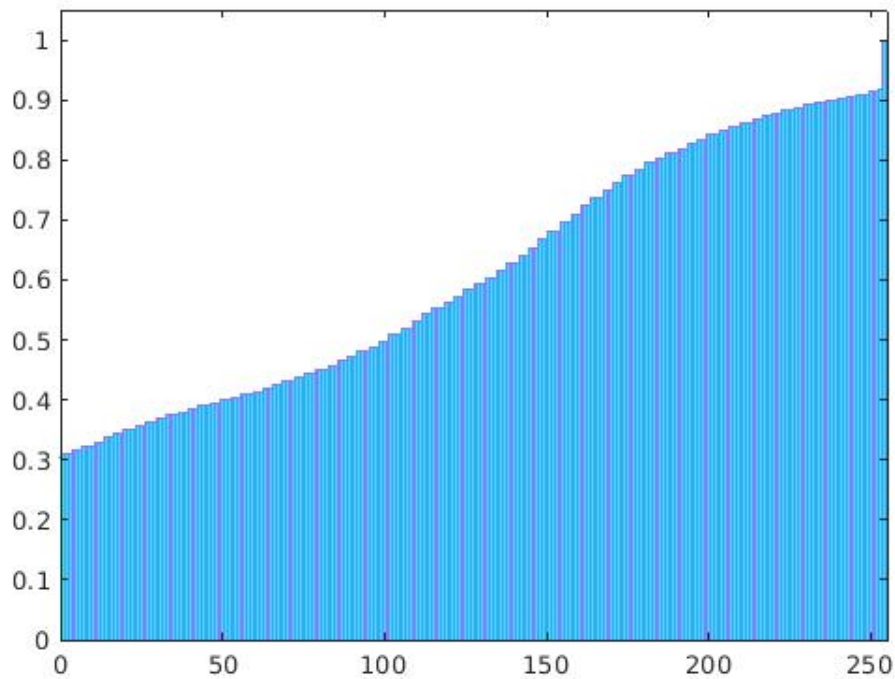
S tem, ko upragujemo sliko pri nekem parametru t v resnici izračunamo koliko sivinskih vrednosti je manjših ali enakih t . S tem v bistvu računamo komulativno ferkvenco. Za isto sliko sem izračunala tudi relativno komulativno ferkvenco, ki je ponazorena v drugem histogramu. Vidimo, da se obliki ujemata. Ker je ena relativna komulativna ferkvenca in ena ni relativna so vrednosti na y osi različne.



Slika 1: Slika, ki jo dobimo če za upragovanje uporabimo parameter $t = 127$



Slika 2: Histogram, ki ponazarja, koliko je elementov $s_g = 0$, glede na paramater t



Slika 3: Histogram relativne komulativne ferkvence slike

```
slika = loadImage('image-512x512-16bit.raw',[512,512], 'int16');
linearna = scaleImage(slika, -0.125, 256);
oknjenje = windowImage(linearna, 1000,500);
prikaz = displayImage(oknjenje, 'oknjenje');

sivniske = linspace(0,1, 256);
stevilo_nicel = zeros(1,256)
for prag = 0:255
    upragovana = thresholdImage(oknjenje, prag);
    indeksi = upragovana <= 0;
    stevilo_nicel(prag+1) = sum(indeksi(:));
end
displayHistogram(stevilo_nicel,sivniske,'Histogram, ki prikazuje število črnih elementov')
[hist, ~, cdf, levels] = computeHistogram(prikaz.CData)
displayHistogram(cdf,levels, 'komulativna ferkvenca');
```

4 Odsekoma nelinearna preslikava

```
function oImage = nonLinearSectionalScaleImage(iImage, iS, oS)
%nonLinearSectionalScaleImage naredi odsekoma nelinearno preslikavo z danimi parametri
%vhodni elementi:
% iImage - vhodna slika v obliki matrike
% iS - seznam x komponent kontrolnih točk
% oS - seznam y komponent kontrolnih točk
%izhodni elementi
% oImage - preslikana slika

oImage = iImage;
for i = 1: 2:length(iS)-2
    Ax = iS(i);% točka A
    Bx = iS(i+1);% točka B
    Cx = iS(i+2); %točka C
    sistem = [Ax^2 Ax 1;
              Bx^2 Bx 1;
              Cx^2 Cx 1]; % izračunane vrednosti za sistem enačb
    Ay = oS(i);
    By = oS(i+1);
    Cy = oS(i+2);
    Y = [Ay;
         By;
         Cy]; %Y komponente enačbe

    koeficienti = sistem\Y %rešimo sistem enačb

    i
    a1 = koeficienti(1)
    b1 = koeficienti(2)
    c1 = koeficienti(3)
    tmp = find(iImage>=Ax & iImage<=Cx);
    %kvadratna funkcija a1*x^2 + b1*x + c1
    oImage(tmp) = a1*iImage(tmp).^2 + b1 * iImage(tmp) +c1;
end
```



Slika 4: Slika, ki jo dobimo, če sivinske vrednosti preslikamo z odsekoma nelinearno preslikavo

Pri odsekoma kvadratični preslikavi dobimo koeficiente polinoma:

	A_i	B_i	C_i
$i = 1$	-0.1344	11.7500	0
$i = 3$	0.0569	-13.0580	760.3830
$i = 3$	-0.0172	8.2531	-731.9297

Tabela 2: Tabela z koeficienti kvadratne funkcije