ROZ1 - Cv. 4 - Detektory hran a ekvalizace histogramu

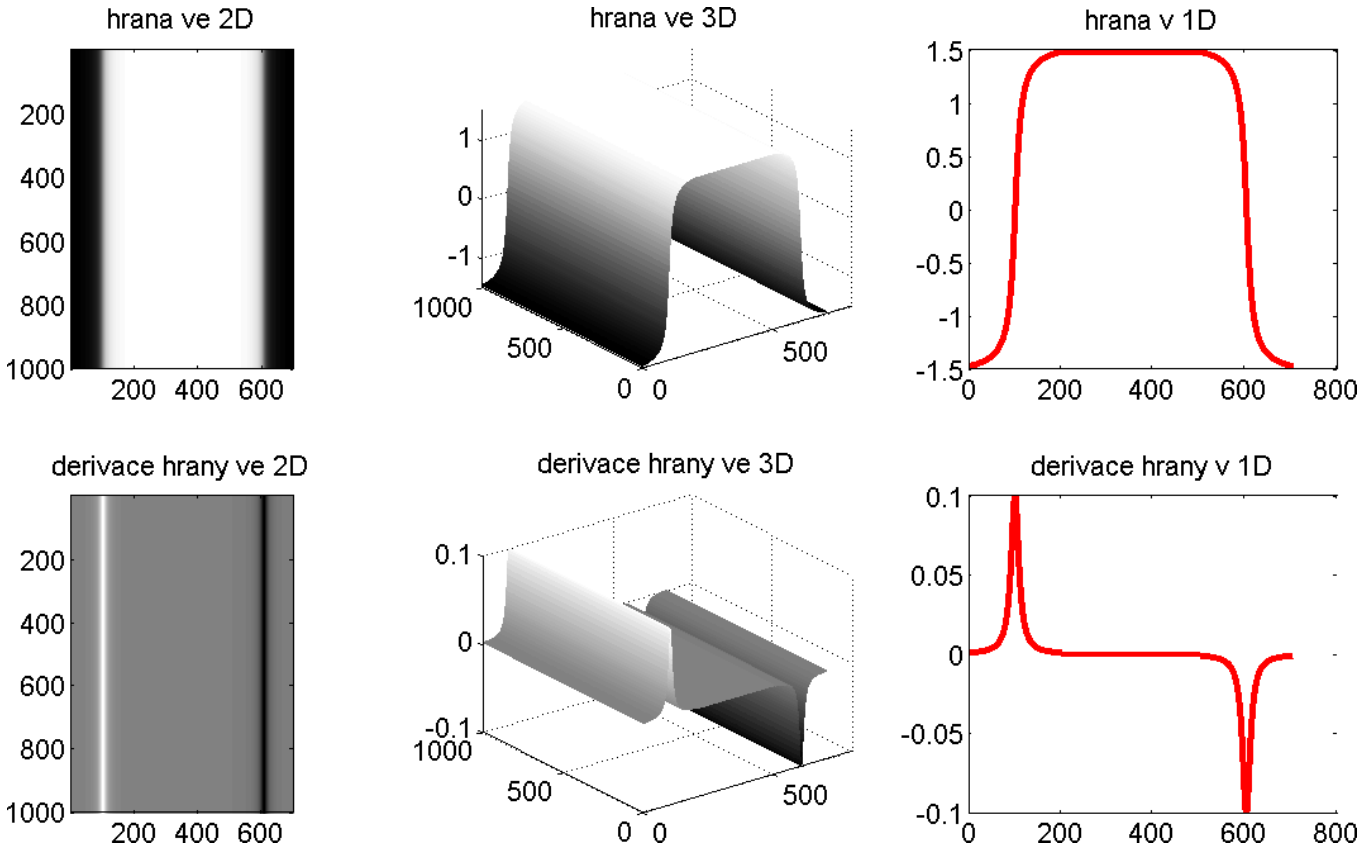
ÚTIA - ZOI

Motivace

- Proč se zabývat v obraze hranami?

Motivace

- Proč se zabývat v obraze hranami?
  - hrany nesou více informace, než místa bez hran
  - jsou důležité pro detekci objektů a segmentaci obrazu
- Co je to hrana v obraze?
  - místo s velkou změnou (gradientem) jasových hodnot pixelů



Obrázek: Vizualizace hrany v obraze

Jaké známe detektory hran?

Jaké známe detektory hran?

- založené na 1. derivaci obrazu:
  - Roberts
  - Sobel
  - Prewitt
  - Kirsch
  - Canny
- založené na 2. derivaci obrazu:

Jaké známe detektory hran?

- založené na 1. derivaci obrazu:
  - Roberts
  - Sobel
  - Prewitt
  - Kirsch
  - Canny
- založené na 2. derivaci obrazu:
  - Marr-Hildreth
- nepracující s derivacemi - Whitening
- pracující ve frekvenční oblasti

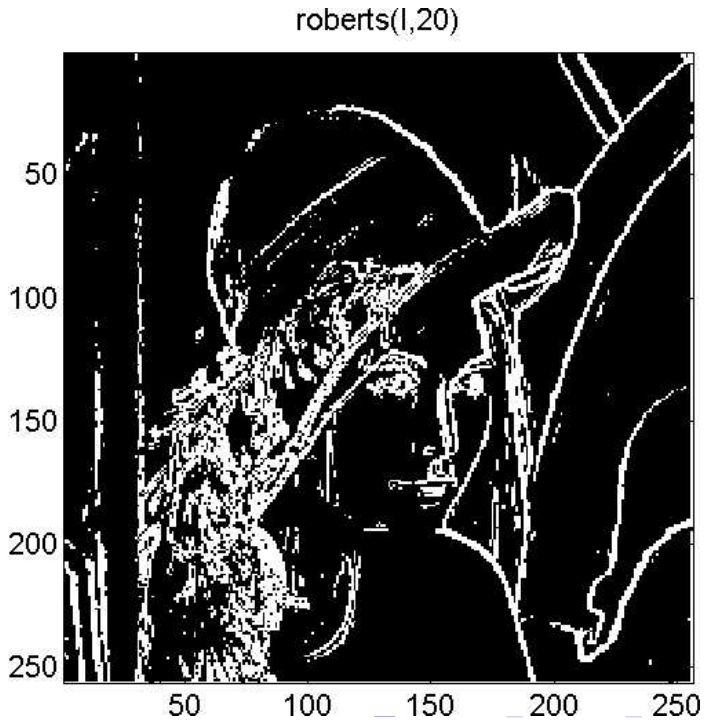
Popište Robertsův detektor:

Popište Robertsův detektor:

- konvoluce s maskami $\begin{pmatrix} -1 & 1 \end{pmatrix}$ a $\begin{pmatrix} -1 \\ 1 \end{pmatrix}$
- porovnám oba směry a beru maximum z nich
- velmi citlivé na šum – kde je šum, jsou všude hrany
- mohu obrázek nejprve trochu rozmazat

Cvičení I. - naprogramujte Robertsův detektor

- ```
  funciton R = roberts(I,Prah)
  % I...obrázek
  % Prah...práh
  ```


  ```

Řešení - Cvičení I.

```
funciton R = roberts(I,Prah)
% I...obrázek
% Práh...práh
M=[-1,1];
R1=abs(conv2(I,M,'same'));
R2=abs(conv2(I,M',’same’));
R=R1 > Prah | R2 > Prah;
```

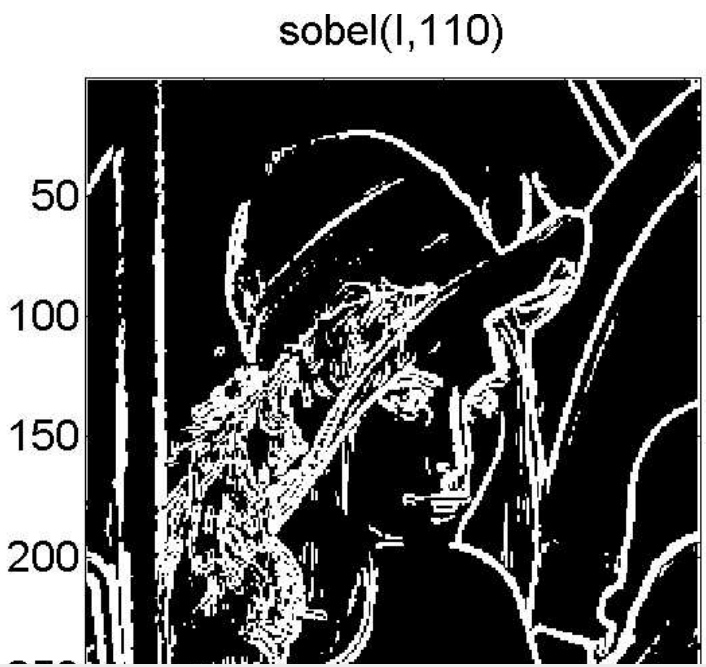
Popište Sobelův detektor:

Popište Sobelův detektor:

- používá masku $\begin{pmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{pmatrix}$
- takových masek je celkem 4 (resp. 8)
- podle maxima vyberu tu největší (je to ta, kde hrana běží ve směru nul)
- jde vlastně o průměrování 1. derivací, kde centrální bod má dvojnásobnou váhu
- robustnější proti šumu, díky velikosti matice

Cvičení II. - naprogramujte Sobelův detektor

- ```
  funciton R = sobel(I,Práh)
  % I...obrázek
  % p...práh
  ```
  - *nápověda* `rot90()`


  ```

Řešení - Cvičení II.

```
funciton R = sobel(I,p)
% I...obrázek
% p...práh
M = [1 2 1; 0 0 0; -1 -2 -1];
R = abs(conv2(I,M,'same')) > Prah;
R = R | (abs(conv2(I,M',’same’)) > Prah);
M = [2 1 0; 1 0 -1; 0 -1 -2];
R = R | (abs(conv2(I,M,’same’)) > Prah);
R = R | (abs(conv2(I,rot90(M),’same’)) > Prah);
```

Prewitt + Kirsch + Robinson- obdobné jako Sobel

- maska Prewittové $\begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{pmatrix}$

Prewitt + Kirsch + Robinson- obdobné jako Sobel

- maska Prewittové $\begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{pmatrix}$
- maska Kirsche $\begin{pmatrix} 3 & 3 & 3 \\ 3 & 0 & 3 \\ -5 & -5 & -5 \end{pmatrix}$

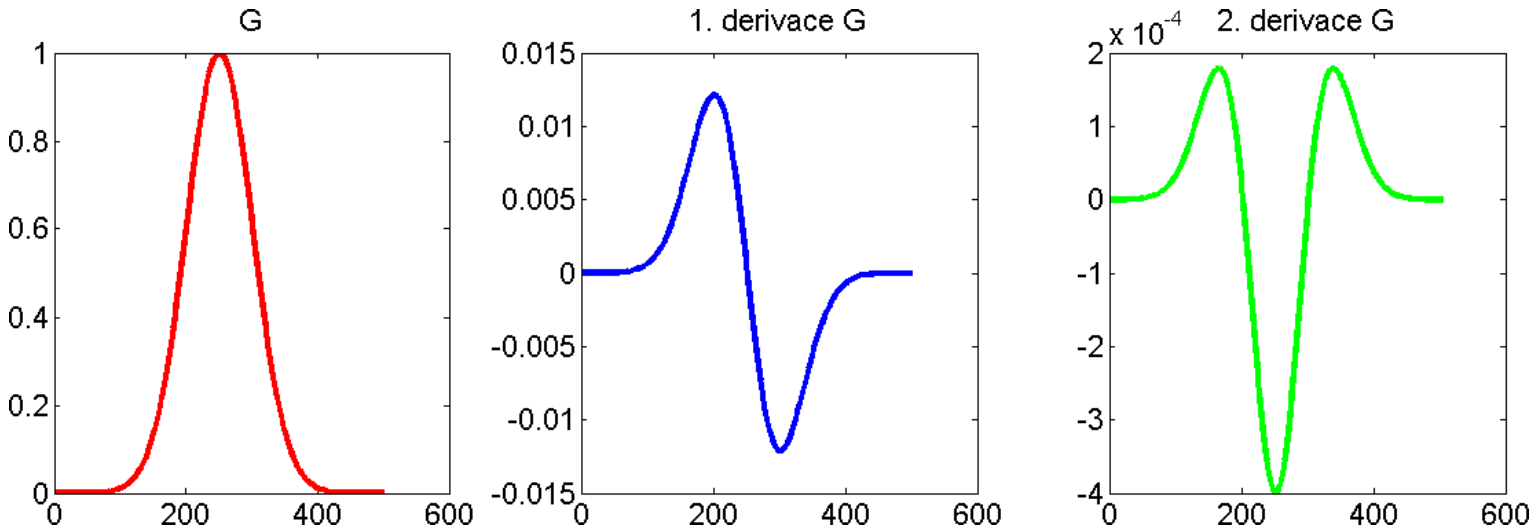
Prewitt + Kirsch + Robinson- obdobné jako Sobel

- maska Prewittové $\begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{pmatrix}$
- maska Kirsche $\begin{pmatrix} 3 & 3 & 3 \\ 3 & 0 & 3 \\ -5 & -5 & -5 \end{pmatrix}$
- maska Robinsona $\begin{pmatrix} 1 & 1 & 1 \\ 1 & -2 & 1 \\ -1 & -1 & -1 \end{pmatrix}$

Popište Marrův detektor:

Popište Marrův detektor:

- **LoG** (Laplacian of Gaussian)
- $\Delta(G * f) = \Delta G * f$
- zero-crossing detection
- hrany mají tendenci se uzavírat do sebe

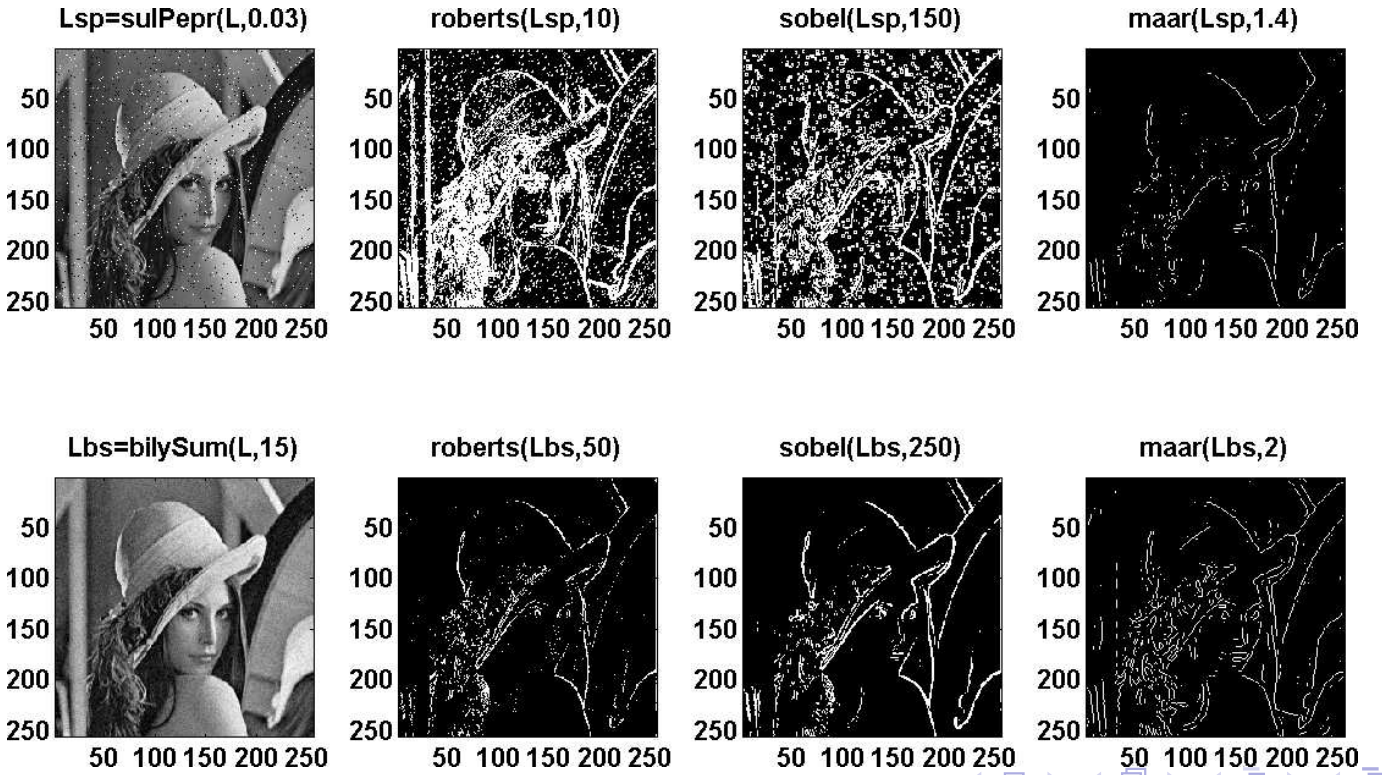


Cvičení III.

- odzkoušejte hranové detektory pro různé šumy
- přiložené skripty `bilySum();` `sulPepr();` `maar();`

Cvičení III.

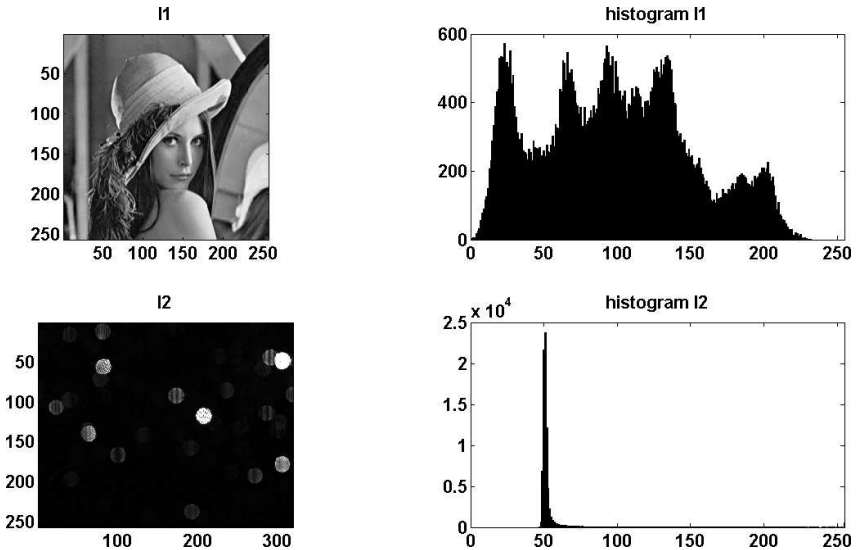
- odzkoušejte hranové detektory pro různé šumy
- přiložené skripty `bilySum();` `sulPepr();` `maar();`



Co to je histogram v obraze:

Co to je histogram v obraze:

- mějme obraz s L úrovněmi jasu
- $H(k) = n_k$, kde n_k je počet pixelů s jasem $k \in < 0, L-1 >$
- Histogram je tedy vektor absolutních četností každého jasu v obraze: $H = [H(0), H(1), ...]$

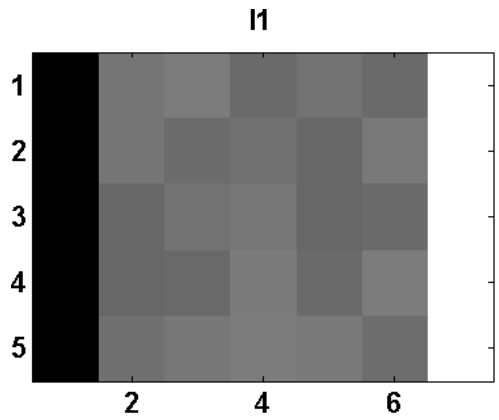


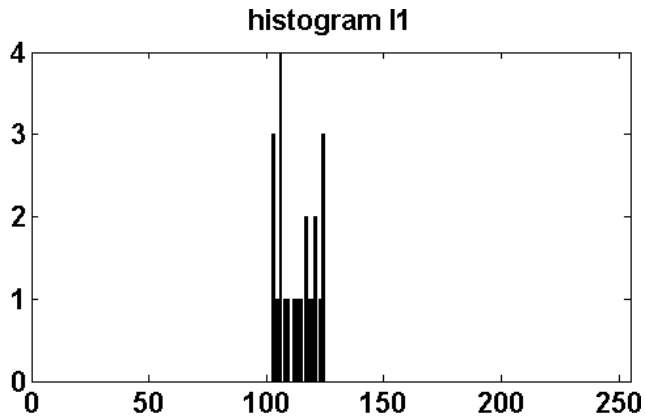
Cvičení IV. - lineární roztažení histogramu

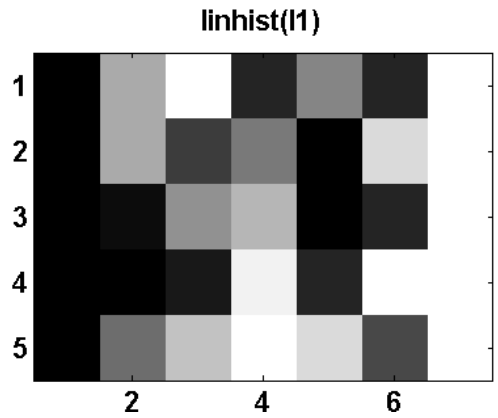
- napište funkci, která "roztáhne"obrázek po celé délce jasu $< 0, 255 >$

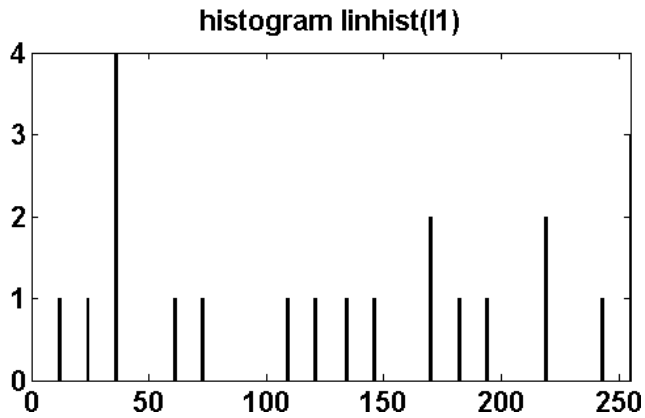
```
funciton R = linhist(I)
% I...obrázek
- nápověda hist(), randi()
```

Řešení - Cvičení IV.









Řešení - Cvičení IV.

- `W=randi([100,125],5,5);`

```
funciton R = linhist(I)
% I...obrázek
R = I-min(I(:));
R = R/max(R(:))*255;
zobr(I);
```

Co to je ekvalizace histogram a jak byste ji provedli:

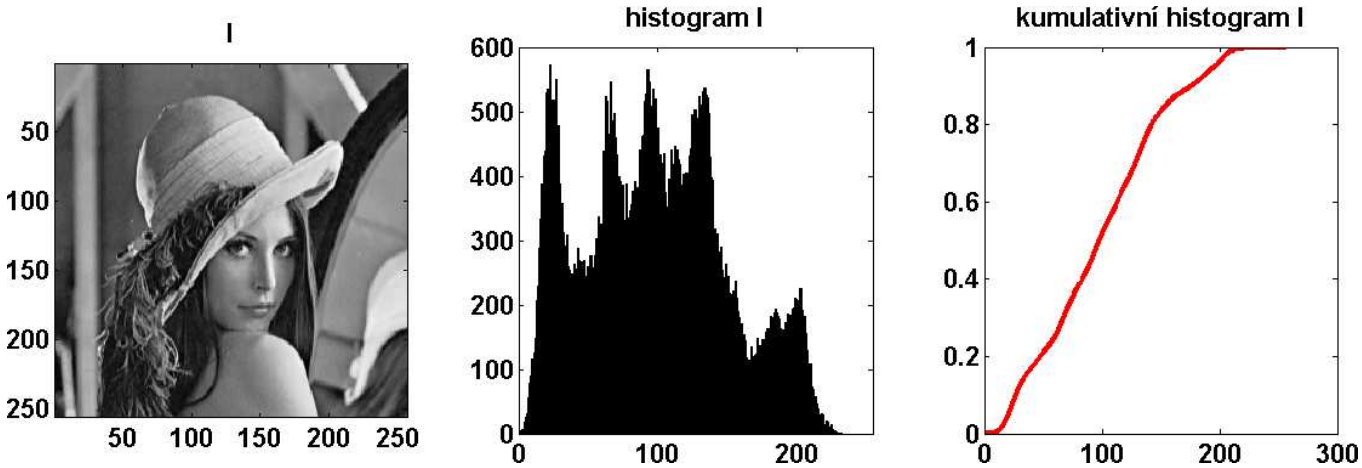
Co to je ekvalizace histogram a jak byste ji provedli:

- jde o tzv. vyrovnání histogramu, resp. zploštění
- jasy se transformují tak, aby každý z nich 'měl pro sebe' takovou část jasové osy, jaké je jeho zastoupení v obrázku
- histogram $\approx$ **hustota pravděpodobnosti**
- kumulativní histogram $\approx$ **distribuční funkce** - co to je?

Kumulativní histogram:

Kumulativní histogram:

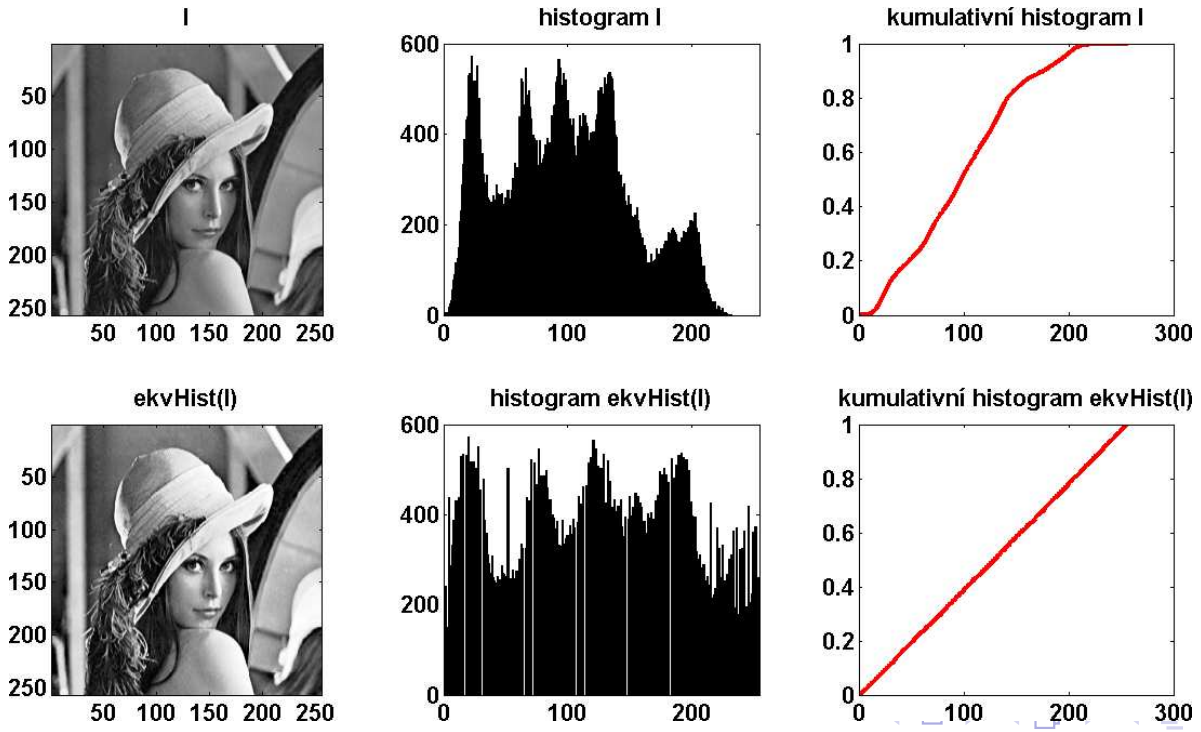
- Nechť n_k je počet pixelů v obraze s úrovní $k \in < 0, L-1 >$
- a $p(x_k) = \frac{n_i}{N}$ četnost výskytu jasu k v tomto obraze (N je počet pixelů celkem)
- kumulativní histogram je pak: $c(k) = \sum_{i=0}^{k} p(x_i)$



K čemu kumulativní histogram?

K čemu kumulativní histogram?

- naše transformace $T(x_k) = c(k) \in < 0, 1 >$
- proto ještě $T(x_k) * (max - min) + min$, u nás $max = 255$ a $min = 0$

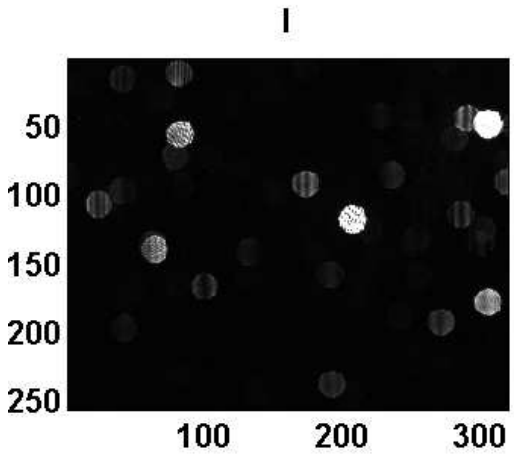


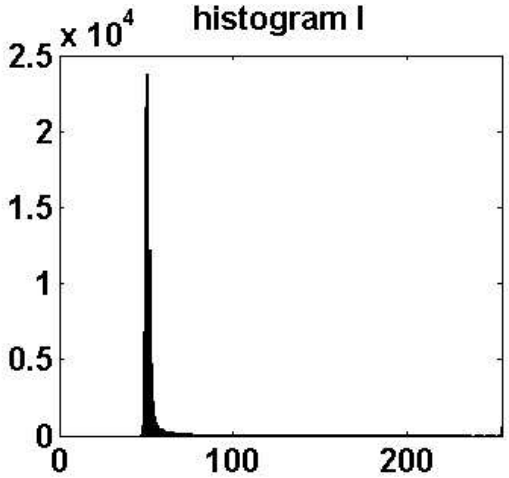
Cvičení V. - ekvalizace histogramu

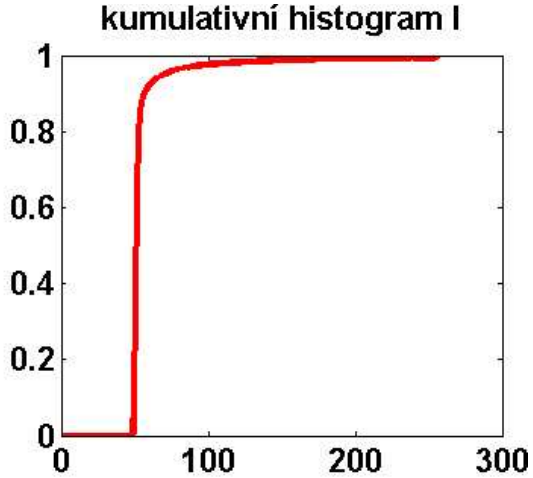
- napište funkci, která ekvalizuje obrázek

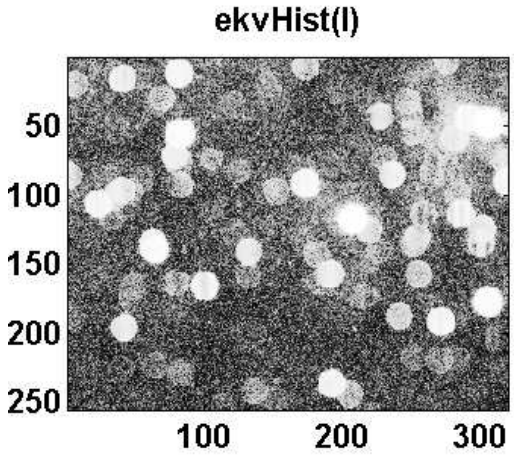
```
funciton R = ekvHist(I)
% I...obrázek
```

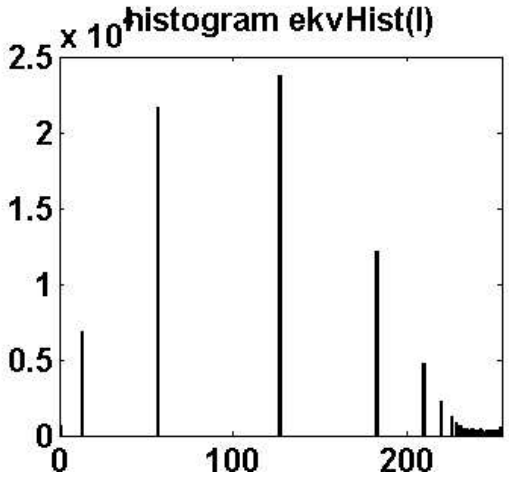
Řešení - Cvičení V.

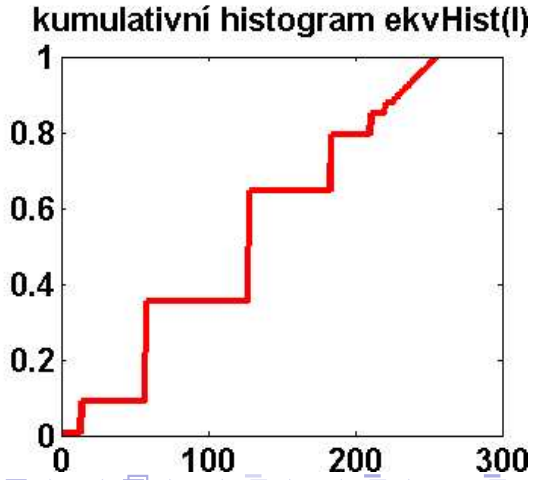












Řešení - Cvičení V.

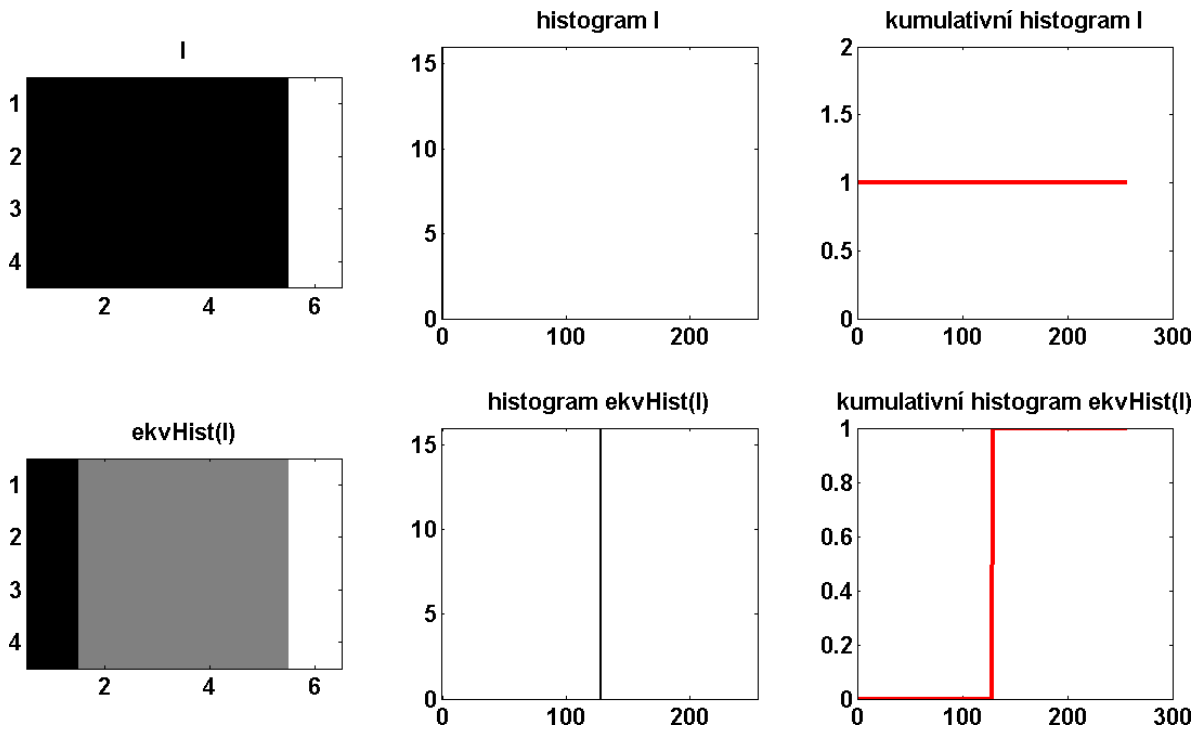
```
funciton R = ekvHist(I)
B = 255;    % intenzita bile barvy
Vel = length(Img(:));    % pocet pixelu v obrazku
R = Img;    % vysledny snimek
S = 0;    % pocet zpracovanych pixelu
for K = 0 : B  % prochazime pixely pres intenzity
  W = (Img == K);    % pixely s intenzitou K
  P = sum (W(:));    % pocet techto pixelu
  R(W) = round((S + P/2) * B / Vel);    % nova
intenzita
  S = S + P;    % pocet zpracovanych pixelu
end
```

Kontrolní otázka:

- Co se stane, když je vstupní obrázek celý černý?

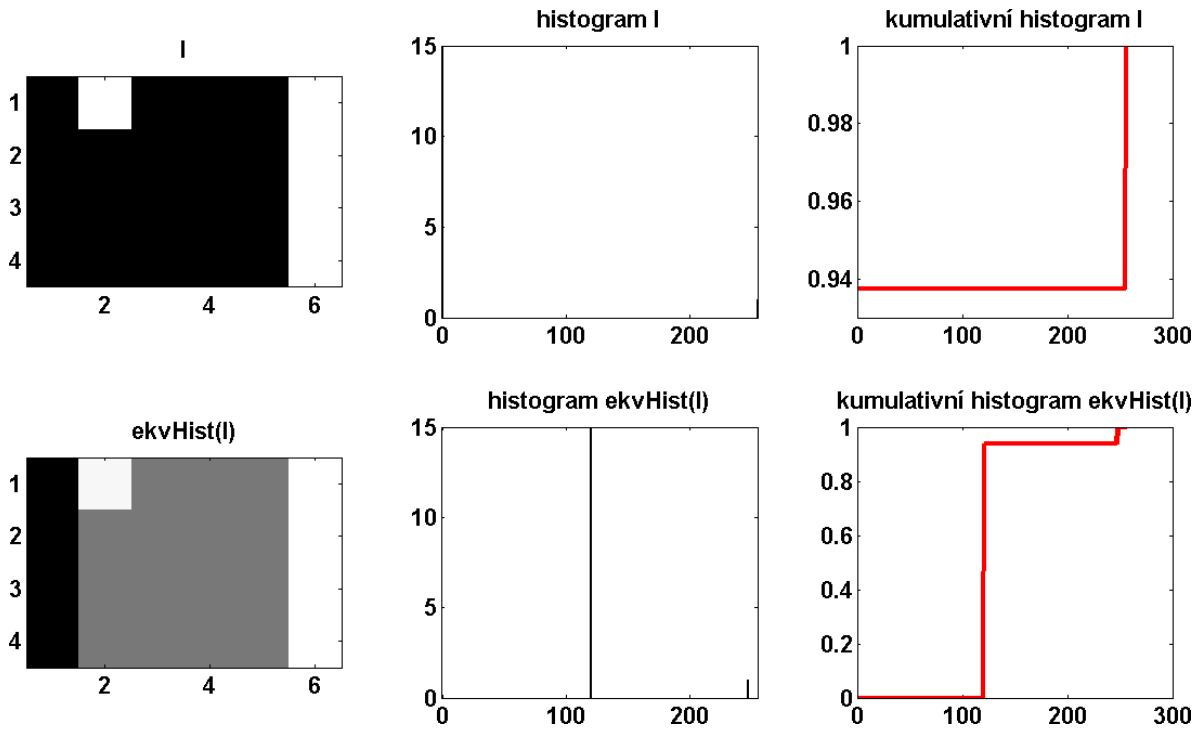
Kontrolní otázka:

- Co se stane, když je vstupní obrázek celý černý?
- Co se stane, když je vstupní obrázek celý černý s bílou tečkou?



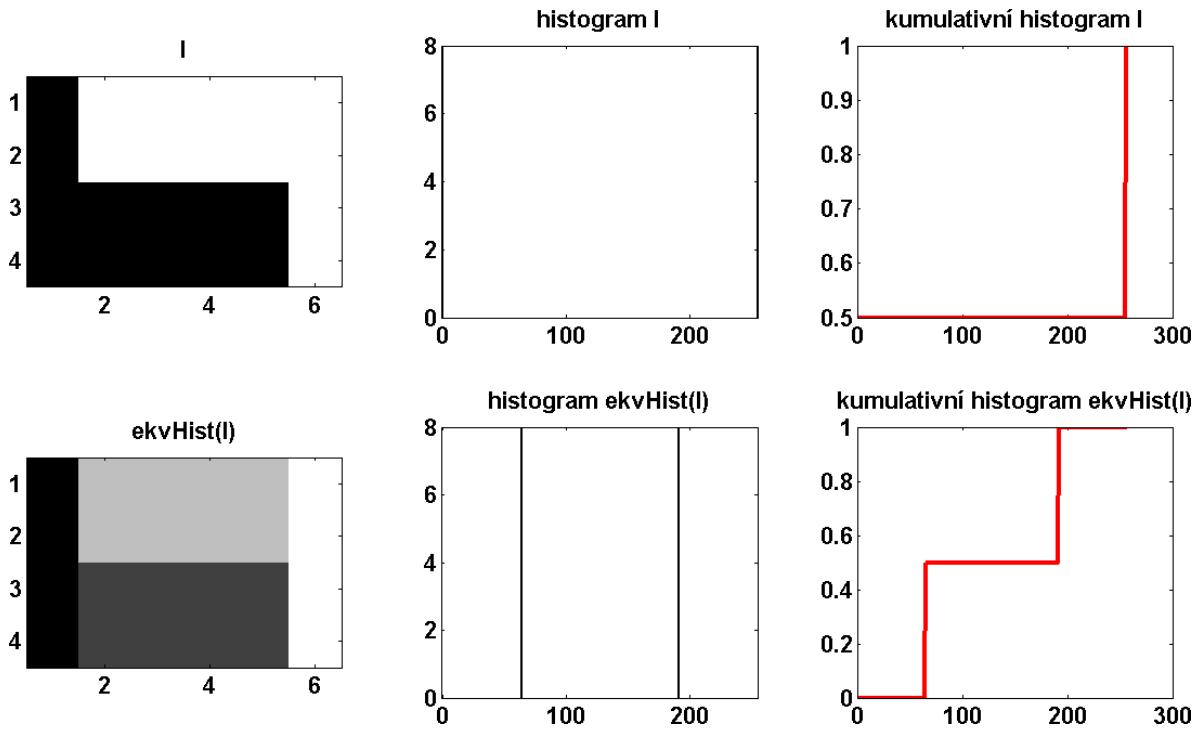
Kontrolní otázka:

- Co se stane, když je vstupní obrázek celý černý?
- Co se stane, když je vstupní obrázek celý černý s bílou tečkou?
- Co se stane, když je vstupní obrázek půl černý půl bílý?



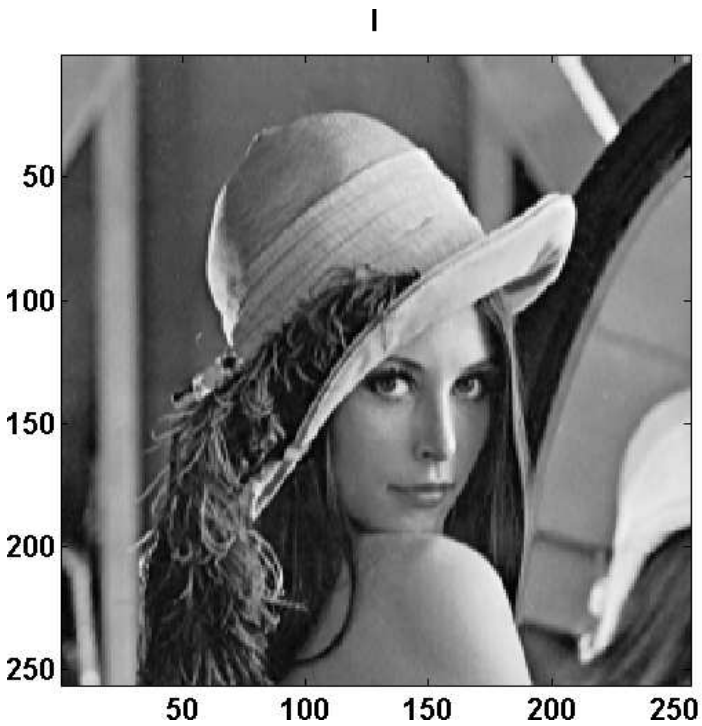
Kontrolní otázka:

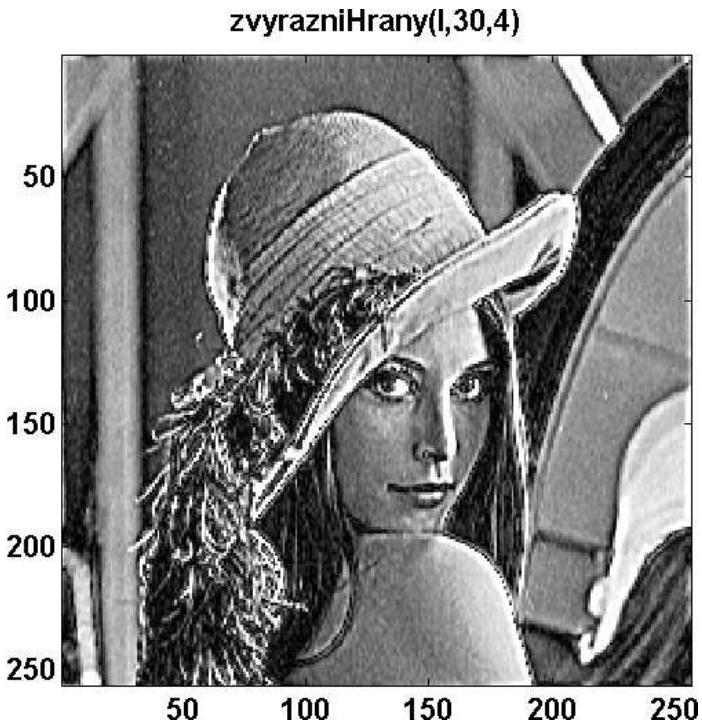
- Co se stane, když je vstupní obrázek celý černý?
- Co se stane, když je vstupní obrázek celý černý s bílou tečkou?
- Co se stane, když je vstupní obrázek půl černý půl bílý?



Bonusový úkol:

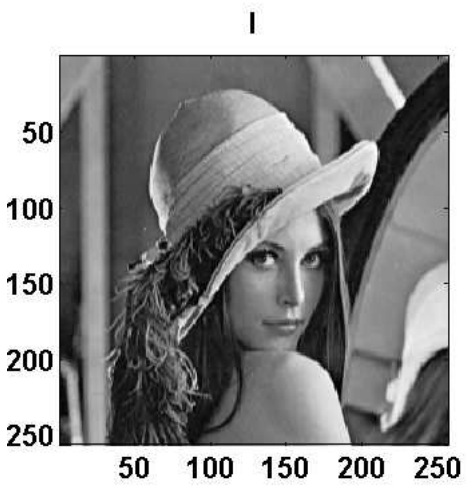
- Zvýraznit na snímku hrany

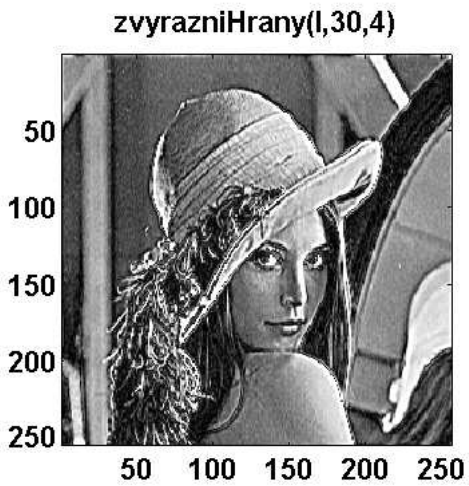


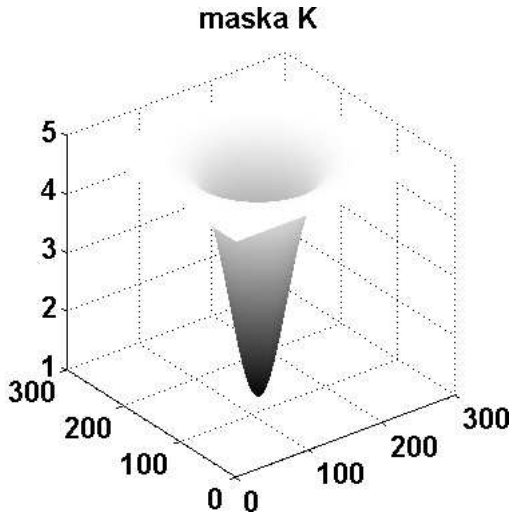


Bonusový úkol:

- Zvýraznit na snímku hrany







Řešení - Bonus

- ```
  funciton R = kruhgauss(R,N)
  % vraci gaussovskou masku o rozptylu R v mat. NxN
  [ X, Y] = meshgrid(-(N-1)/2:(N-1)/2,...
     -(N-1)/2:(N-1)/2);
  K=exp(-0.5*(X.^2+Y.^2)/R.^2);
  ```
  ```

Řešení - Bonus

```
funciton R = zvyrazniHrany(I,prah,coef)
F=fft2(I);
mi=min(I(:));
ma=max(I(:));
K=(-kruhgauss(prah,size(I,1))+1)*coef+1;
J=abs(ifft2(F.*fftshift(K)));
J(J<mi)=mi;
J(J>ma)=ma;
%--------------
mesh(K);
zobr(J);
```

Co jsme se dnes naučili:

- detekovat hrany
- pracovat s histogramem obrázku
- roztáhnout ho a ekvalizovat

KONEC

Děkuji za pozornost !