

Príznaky textúry

Cvičenia z Počítačového Videnia 2

Zuzana Haladová

Príznaky

- Úloha 1
- Gaborove filtre
- Matica opakovaných výskytov
- Houghova transformácia

Úloha 1

Cvičenia z Počítačového Videnia 2

Zuzana Haladová

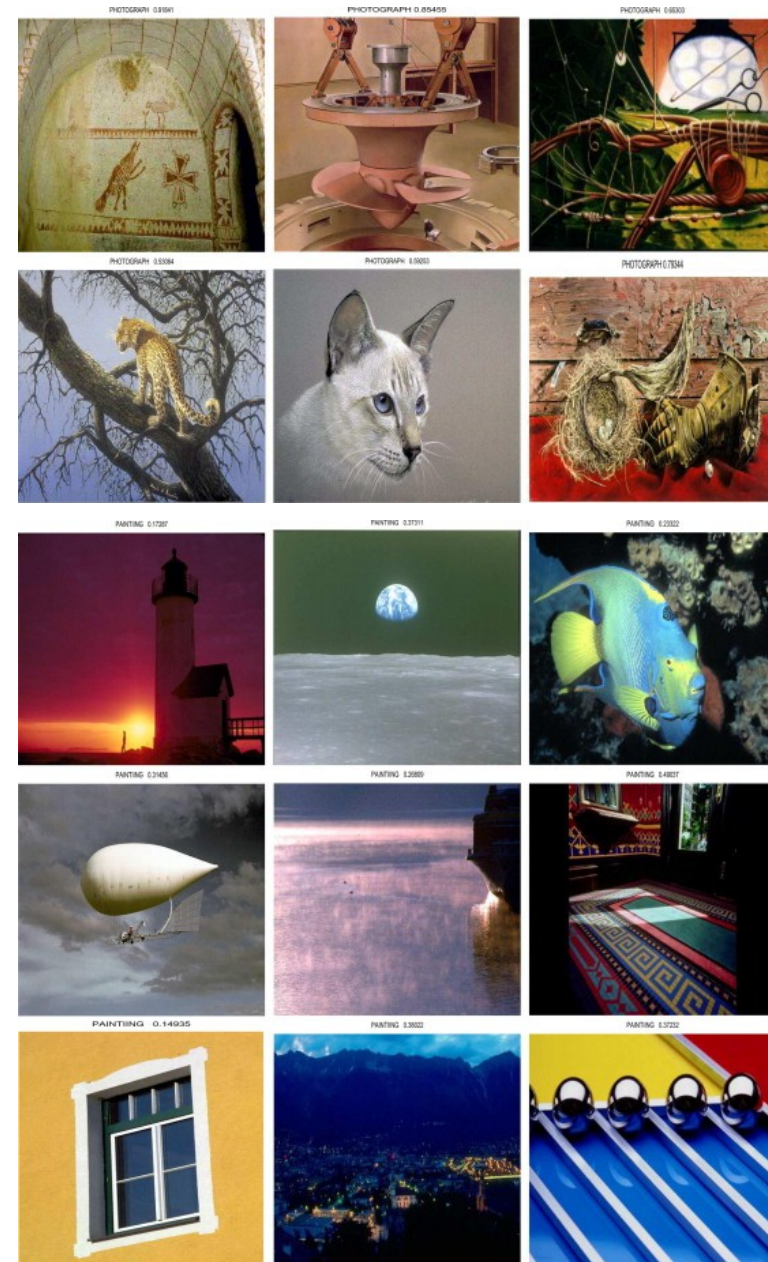
Photographs vs. Paintings

Rozlišovanie malieb a fotografií

6000 obrázkov – 6000 malieb

Florin Cutzu, Riad Hammoud, Alex Leyk:
Distinguishing paintings from photographs
2005

Množstvo nízkoúrovňových príznakov



Photographs vs. Paintings

Zmena farby medzi pixlami je výraznejšia v maľbách

Maľby obsahujú hlavne vysoko saturované farby.

Odstránením farby z obrázku zachováme väčšinu hrán vo fotografií, ale odstránime väčšinu hrán z maľby.

Väčšina hrán na **fotografiách** je spôsobená zmenou **intensity**, interakciou svetla a povrchov s rôznou odrazivosťou.

Hrany v **maľbách** sú väčšinou na prechodoch medzi rôznymi **farbami** ktoré reprezentujú rozličné objekty a oblasti

Uloha 1

Vytvorte aplikáciu ktorá umožní zadať obrázok a vypíše či je obrázok fotografia alebo maľba.

- 1)Využite príznaky z tejto prezentácie (3 príznaky), identifikujte o aké príznaky ide a implementujte ich
- 2)Určte prahy pre klasifikáciu jednotlivých príznakov podľa obrázkov z trénovacej databázy
- 3)Vytvorte GUI
- 4)Pre vstupný obrázok aplikácia vyhodnotí či ide o maľbu alebo fotografiu

Do 20.3 23:59 za 10 bodov

Uloha 1

Príznaky:

Maľby obsahujú viac pixlov ktoré sú vysoko saturované (saturované na viac ako 75 %) ako fotografie (pozn. HSV)

Maľby obsahujú viac unikátnych farieb (pozn. rgb trojice)

Odstránením farby z obrázka odstránime väčšinu hrán z maľby, ale zachováme väčšinu hrán vo fotografií

Uloha 1

Príznak 3 (Pomôcka):

The intensity edges were obtained by converting the image to gray-scale and applying the Canny edge detector.

Image intensity information was removed by dividing the R, G, and B image components by the image intensity at each pixel, resulting in normalized RGB components: $R_n = R/I$, $G_n = G/I$, $B_n = B/I$

Where $I = 0.3R + 0.6G + 0.1B$ is image intensity.

The color edges of the resulting “intensity-free” color image were determined applying the Canny edge detector to the three color channels and fusing the resulting edges.

Uloha 1

Klasifikácia:

Určte prahy pre klasifikáciu jednotlivých príznakov podľa obrázkov z databázy.

Vypočítajte hodnoty jednotlivých príznakov pre všetky obrázky z databázy
Určte prah pre každý príznak, ktorý určí či ide o maľbu alebo fotografiu

Príklad: určite medián hodnôt príznaku pre Maľby, medián pre Fotografie a
zvoľte priemer týchto hodnôt ako prah

Uloha 1

Vytvorte GUI, ktoré by malo obsahovať:

Možnosť vybrať obrázok a vyhodnotiť o aký typ ide

Možnosť použiť vlastnú databázu z ktorej sa potom zistia prahy.

(Stačí keď umožní načítať adresáry Painting a Photograph a vypíše prahy, napr. Stlačenie tlačidla = výpočet prahov,)

Príznaky textúry

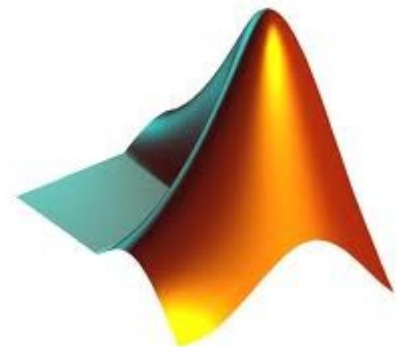
Cvičenia z Počítačového Videnia 2

Zuzana Haladová

Hough

Houghova transformácia- ak chceme detekovať oblasti zo známym tvarom hranice

- Vieme detekovať priamky aj krivky, ak je známe ich analytické vyjadrenie
- Metóda je robustná aj pre zašumené objekty



Hough

Houghova transformácia pre priamky

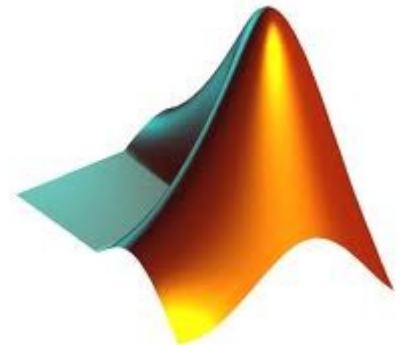
- Priamku vieme (v 2D) vyjadriť

$$y = mx + b$$

- Houghova trans. reprezentuje priamku pomocou b , m resp. r , θ pričom

- r je najmenšia vzdialenosť od počiatku k priamke

- θ je uhol medzi x a priamkou



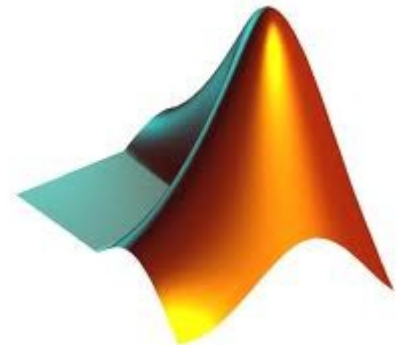
Hough

Vyjadrenie priamky pomocou r a θ je:

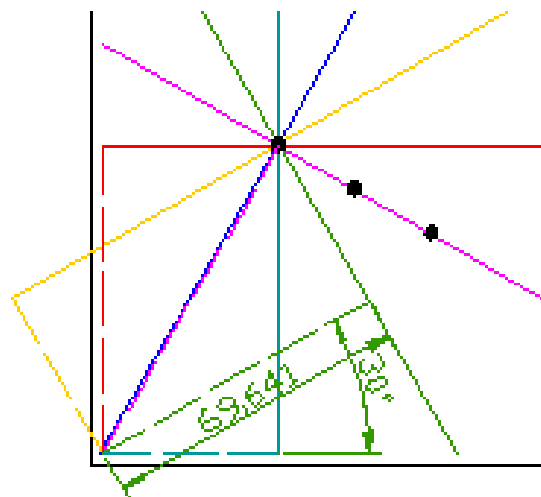
$$y = \left(-\frac{\cos \Theta}{\sin \Theta}\right)x + \left(\frac{r}{\sin \Theta}\right)$$

HT pracuje na bin. obrázku nasledovne:

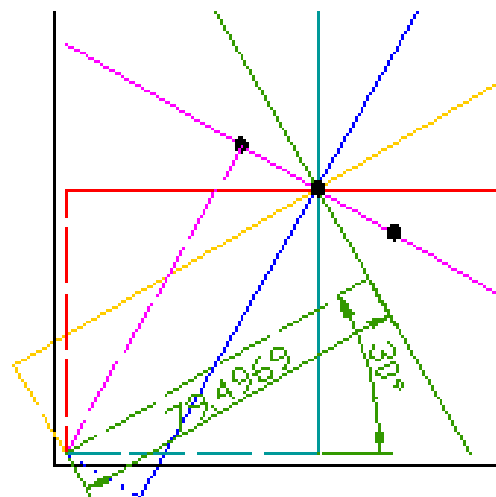
- Pre každý bod obrázku a pre každú priamku ním prechádzajúcu vypočíta r a θ tejto priamky.



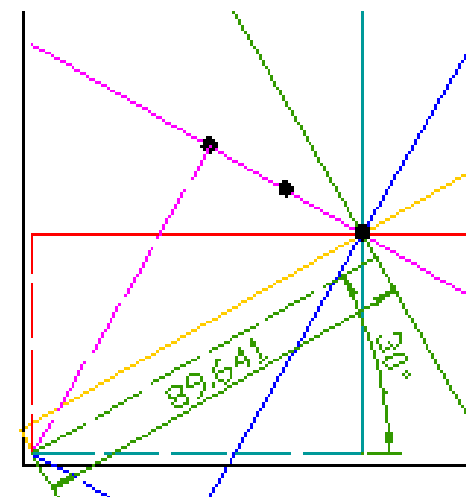
Hough



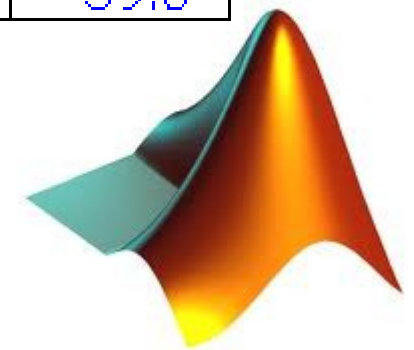
Angle	Dist.
0	40
30	69.6
60	81.2
90	70
120	40.6
150	0.4



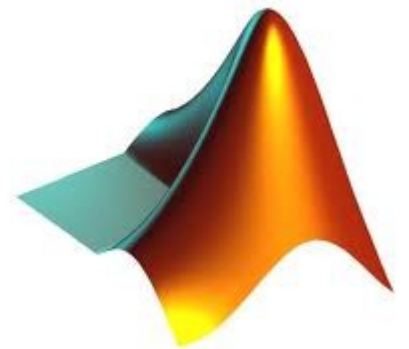
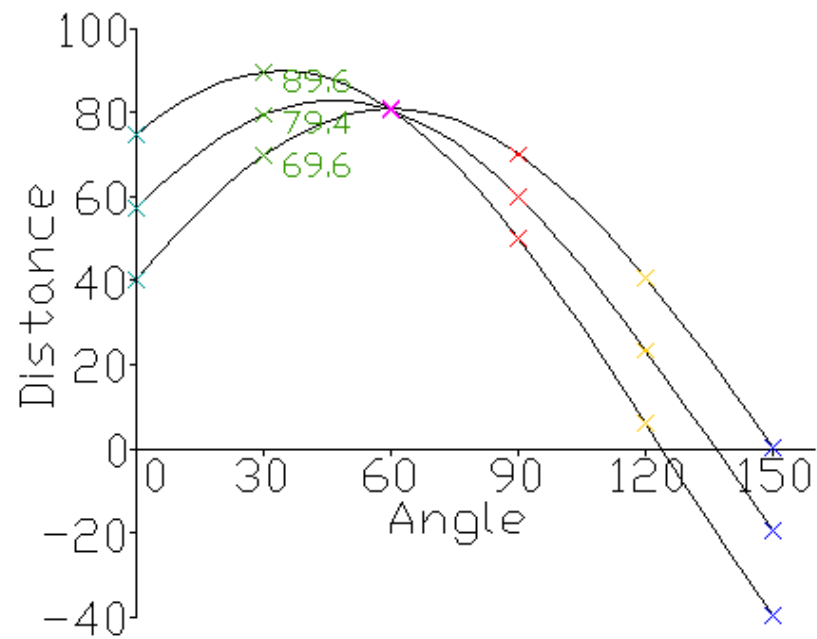
Angle	Dist.
0	57.1
30	79.5
60	80.5
90	60
120	23.4
150	-19.5



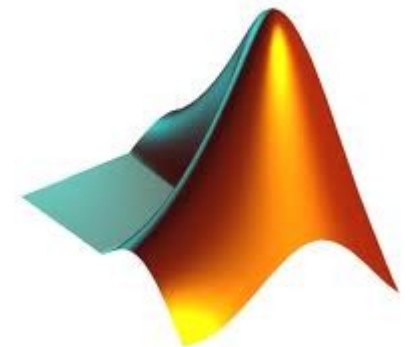
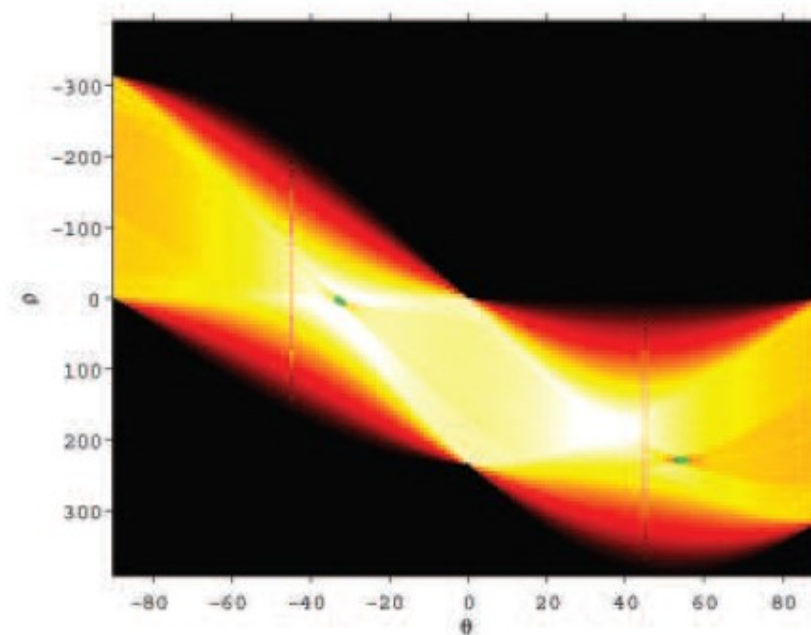
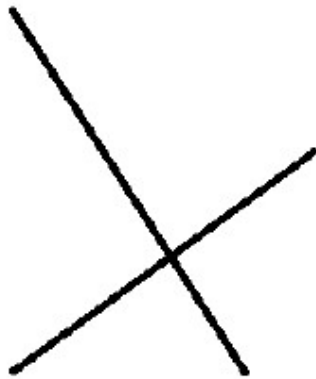
Angle	Dist.
0	74.6
30	89.6
60	80.6
90	50
120	6.0
150	-39.6



Hough



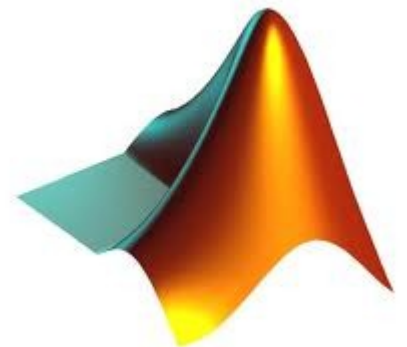
Hough



Hough

V MATLAB (Image Processing Toolbox)

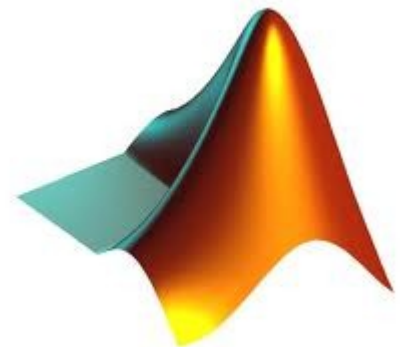
```
I=imread("Ciary.JPG"); IG =rgb2gray(I);  
IE=edge(IG,'Canny');  
[H,T,R]=hough(IE);  
imshow(imadjust(mat2gray(H)),'XData',T,'YData',R);  
colormap(hot);
```



Hough

$[H,T,R]=\text{hough}(IE);$

- H – matica Houghovej transformácie
- T je pole hodnôt θ a R je pole r nad ktorými je vyjadrená H
- Riadky a stĺpce H zodpovedajú r a θ
- hodnota na pozícii (1,2) zodpovedá počtu bodov ležiacich na priamke z 1. r a 2. θ z polí R a T



Hough

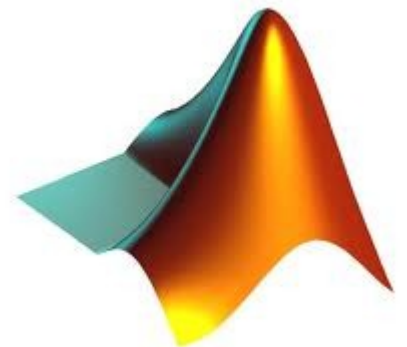
`[H,T,R]=hough(IE);`

- `peaks = houghpeaks(H, numpeaks)`

- `peaks` je pole $Q \times 2$ kde Q

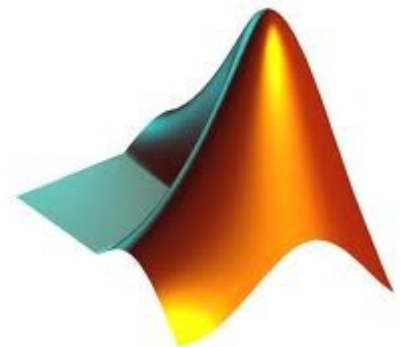
`<0,numpeaks>`

a obsahuje x,y hodnoty pre `peaks` v `H`



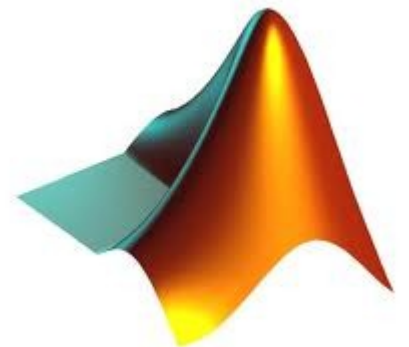
Hough

- `lines = houghlines(IE, T,R, peaks)`
- `lines` je štruktúrované pole ktoré pre každú nájdenú úsečku obsahuje:
 - `point1` x,y hodnoty konca úsečky
 - `point2` x,y hodnoty konca úsečky
 - `theta` uhol v stupňoch (z matice `H`)
 - `rho` hodnota(z matice `H`)



Hough

```
lines = houghlines(IE, T, R, peaks)
- figure, imshow(IE), hold on
- for k = 1:length(lines)
- xy = [lines(k).point1; lines(k).point2];
- plot(xy(:,1),xy(:,2),'Color','green');
- end
```

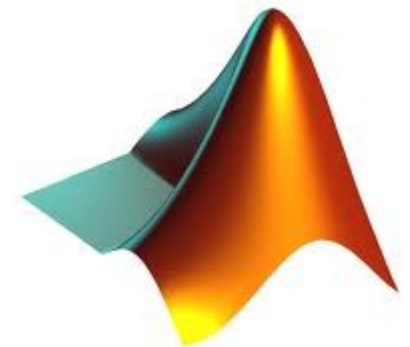
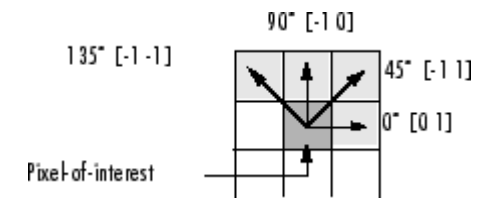
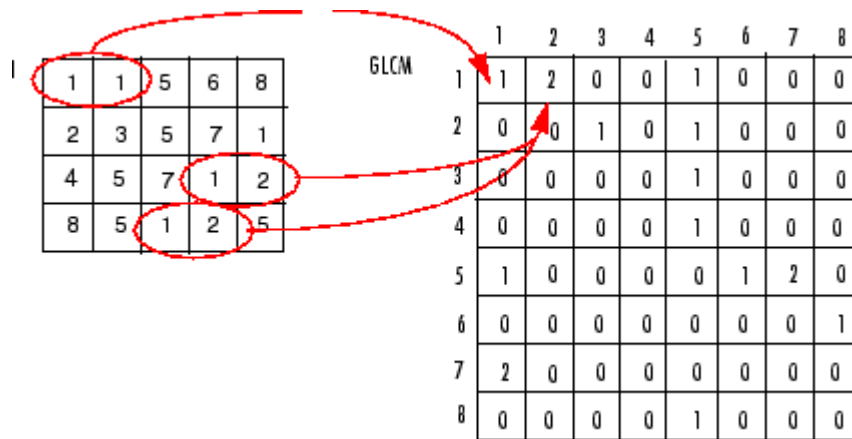


Matica opakovaných výskytov

`glcm = graycomatrix(I)`

I- šedoúrovňový/binárny obrázok (8/2)

`graycomatrix(I, 'offset', [0 1], 'Symmetric', false)`



Matica opakovaných výskytov

```
glcm = graycomatrix(I)
```

```
stats = graycoprops(glcm, properties)
```

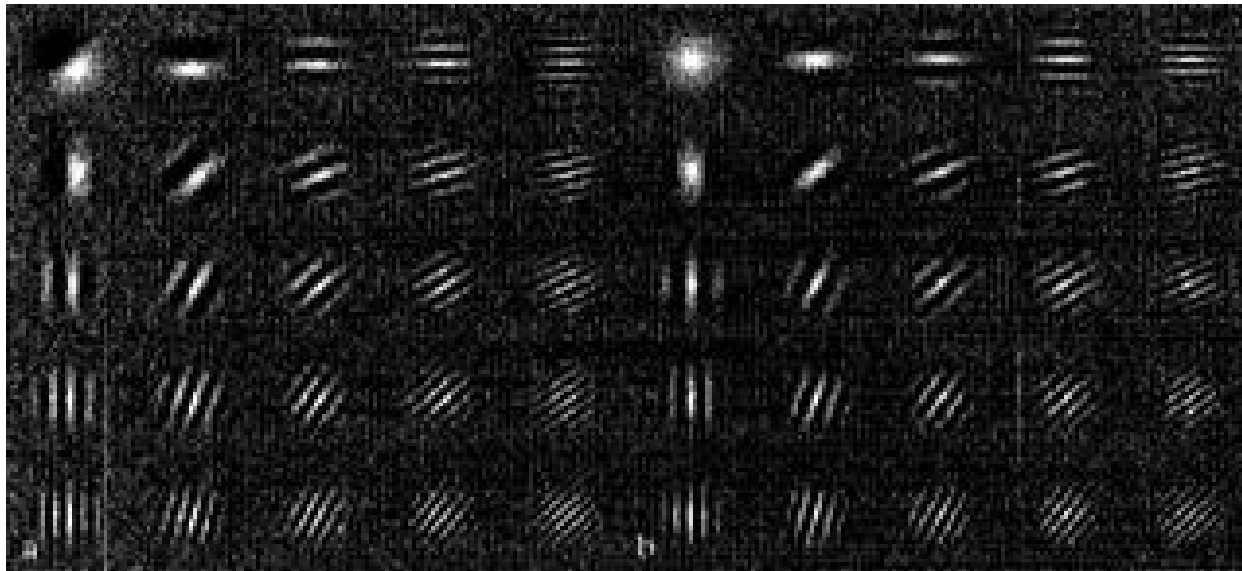
```
stats = graycoprops(glcm,{'contrast','homogeneity'})
```

Property	Description	Formula
'Contrast'	Returns a measure of the intensity contrast between a pixel and its neighbor over the whole image. Range = $[0 \text{ (size(GLCM,1)-1)^2}]$ Contrast is 0 for a constant image.	$\sum_{i,j} i-j ^2 p(i,j)$
'Correlation'	Returns a measure of how correlated a pixel is to its neighbor over the whole image. Range = $[-1 \ 1]$ Correlation is 1 or -1 for a perfectly positively or negatively correlated image. Correlation is NaN for a constant image.	$\sum_{i,j} \frac{(i-\mu_i)(j-\mu_j) p(i,j)}{\sigma_i \sigma_j}$
'Energy'	Returns the sum of squared elements in the GLCM. Range = $[0 \ 1]$ Energy is 1 for a constant image.	$\sum_{i,j} p(i,j)^2$
'Homogeneity'	Returns a value that measures the closeness of the distribution of elements in the GLCM to the GLCM diagonal. Range = $[0 \ 1]$ Homogeneity is 1 for a diagonal GLCM.	$\sum_{i,j} \frac{p(i,j)}{1+ i-j }$

Textúra

Gaborove filtre

The Gabor filter is basically a Gaussian (with variances s_x and s_y along x and y -axes respectively) modulated by a complex sinusoid (with centre frequencies U and V along x and y -axes respectively)



Textúra

```
I=imread('Sandeman.jpg')  
[G,gabout] = gaborfilter1(rgb2gray(I),2,4,16,pi/3);  
figure,imshow(uint8(gabout));  
surf(G)
```

