University of Rochester. ECE 449 Machine Vision.

Tiecheng Su, Dept. of Electrical Engineering

HW4. Submission: 28 March 2016

%% Initialization

close all; clear; clc

%% Removing small regions and regions with aspect ratio less than 3

I = imread('E:\2016Spring\MV\HW4\Shapes\_blurred\_removed.png');

I = flipud(I);

%Since y is counting from the bottom up, we better filip the image

%extracting boundary points

B = bwboundaries(I,'noholes');

I\_b = zeros(size(I));

for k = 1:length(B)

boundary = B{k};

for i=1:size(boundary,1)

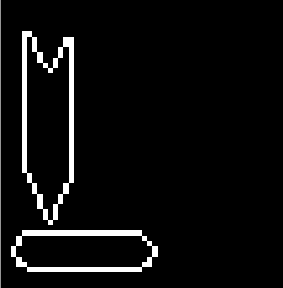
I\_b(boundary(i,1),boundary(i,2))=1;

end

end

figure;

imshow(flipud(I\_b));

E:\2016Spring\MV\HW4\Shapes_blurred_removed.png

%% Use polar representation

close all

thetafrequency = 0.01;

[rows,cols] = size(I);

rho\_limit = norm([rows cols]);

%quantize angle and offset

rho = -rho\_limit:0.4:rho\_limit;

theta = 0:thetafrequency:pi;

houghspace = zeros(numel(rho),numel(theta));

%pre-compute the sin and cos values

cosine = (0:cols-1)'\*cos(theta);

sine = (0:rows-1)'\*sin(theta);

[y\_index,x\_index] = find(I\_b);

accumulator = zeros(numel(y\_index),numel(theta));

accumulator(1:numel(y\_index),:) = cosine(x\_index,:) + sine(y\_index,:);

for i=1:numel(theta)

houghspace(:,i) = hist(accumulator(:,i),rho);

end

figure;

pcolor(theta,rho,houghspace);

shading flat;

hold on

peaks = houghpeaks(houghspace,4);

title('Hough Transform');

xlabel('theta(radians)');

ylabel('rho(pixels)');

plot(theta(peaks(:,2)),rho(peaks(:,1)),'s','color','white')

hold off

fprintf('The corresponding thetas are: \n');

fprintf('%.2f\n',theta(peaks(:,2))\*180/pi);

fprintf('The corresponding rhos are:\n');

fprintf('%.2f\n',rho(peaks(:,1)));

%The length of each line should be the peaks values

fprintf('The corresponding lengths are:\n')

for i=1:4

length = houghspace(peaks(i,1),peaks(i,2));

fprintf('%d\n',length(1));

end

The corresponding lengths are:

30

29

23

22

%The correspondence of peaks

figure;

imshow(I\_b);

hold on

for i=1:2

line([0 rho(peaks(i,1))\*cos(theta(peaks(i,2)))],...

[20\*i rho(peaks(i,1))\*sin(theta(peaks(i,2)))+20\*i],'Color','r','LineWidth',3);

text(2,20\*i,num2str(i),'Color','w','Fontsize',15,'Fontweight','bold');

end

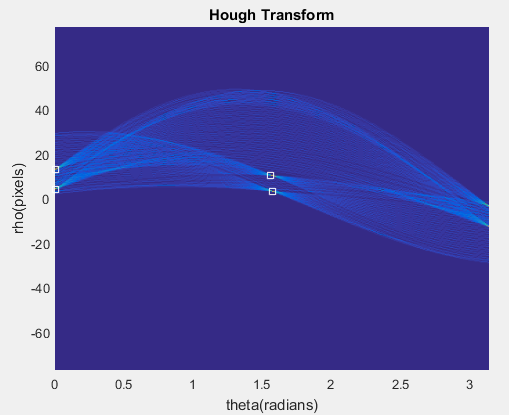
for i=3:4

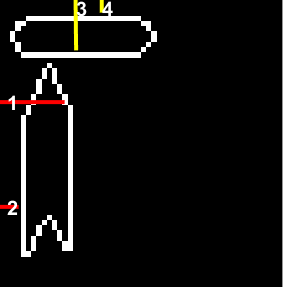
line([i\*5 rho(peaks(i,1))\*cos(theta(peaks(i,2)))+i\*5],...

[0 rho(peaks(i,1))\*sin(theta(peaks(i,2)))],'Color','y','LineWidth',3);

text(i\*5,2,num2str(i),'Color','w','Fontsize',15,'Fontweight','bold');

end



The corresponding thetas are:

0.00

0.00

89.38

89.95

The corresponding rhos are:

12.92

4.12

10.12

2.92

In order to find the peaks, we just need to find out the local maximum. Considering detect the straight lines, smoothing doesn’t help a lot.

The polar representation can be equally quantize compare to the standard one.