Image Processing Project #6

0510894 電機 4D 翁紹恩

I. Source codes (With Matlab)

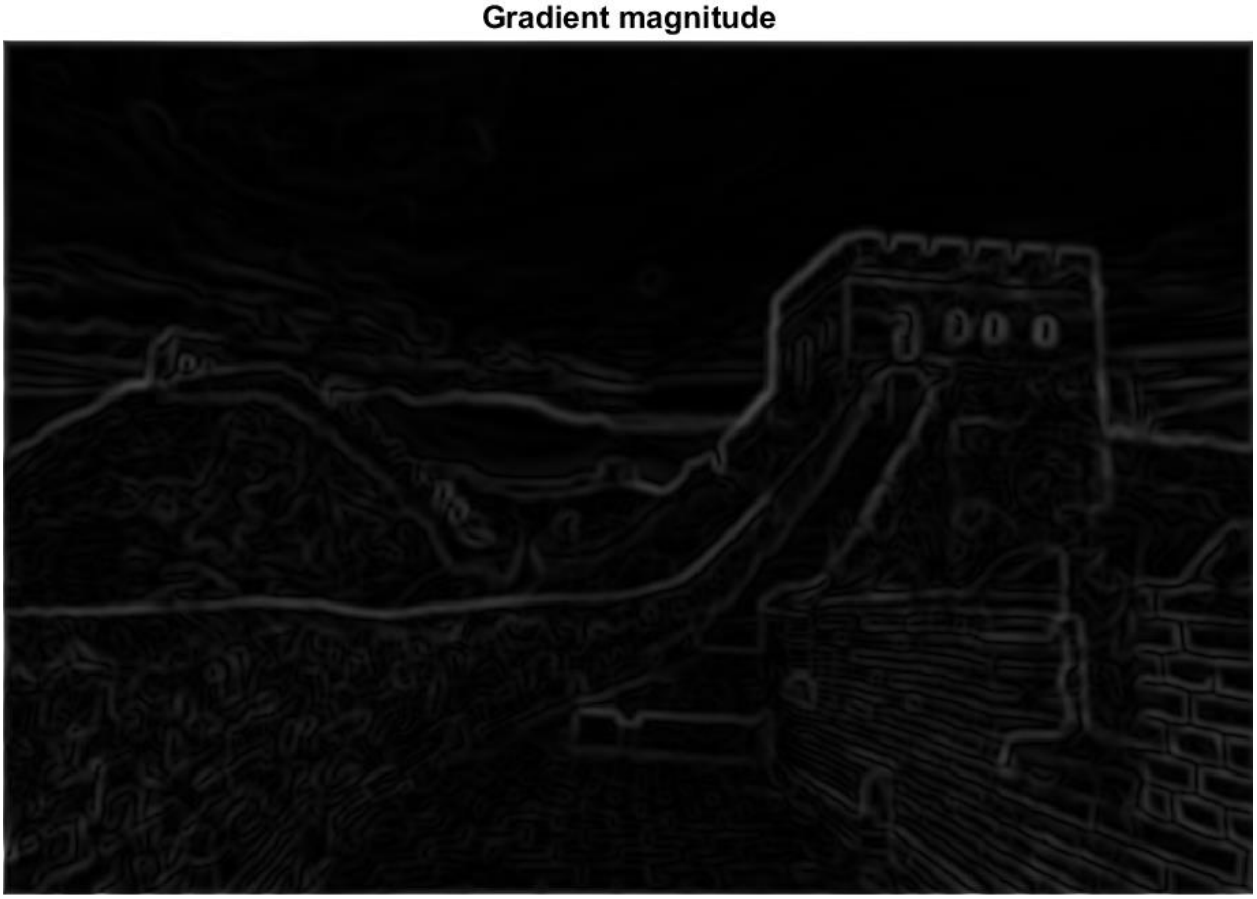
```
%%%%%%%%%%% load image %%%%%%%%%%%%%%%%
Img=im2double(imread('image-pj6(Canny).tif'));
%%%%%%%%%%% gaussian & sobel %%%%%%%%%%%%%%%%
[w,h] = size(Img);
sig = min([w,h]) * 0.005;
x = floor(-3*sig):ceil(3*sig);
y = floor(-3*sig):ceil(3*sig);
y = repmat(y,19,1);
x = repmat(x',1,19);
G = exp((-0.5*x.^2 -0.5*y.^2)./sig^2);
sobel_x = [-1,-2,-1;0,0,0;1,2,1];
sobel_y = [-1,0,1;-2,0,2;-1,0,1];
%%%%%%%%%%% apply filter %%%%%%%%%%%%%%%%
sxg = conv2(Img, G ,'same');
syg = conv2(Img, G ,'same');
%%%%%%%%%%% scale intensity %%%%%%%%%%%%%%%%
sxg = sxg./sum(sum(G));
syg = syg./sum(sum(G));
sx = conv2(sxg, sobel_x ,'same');
sy = conv2(syg, sobel_y ,'same');
sx = -1.*sx;    % inverse because conv2 will turn opposite 180 degrees direction
sy = -1.*sy;
%%%%%%% compute the gradient and the magnitude %%%%%%%
Mag = sqrt( sx.^2 + sy.^2 );
Angle = atan2(sy, sx) * (180.0/pi);
figure;
imshow(Mag,[])
title('Gradient magnitude')
saveas(gcf,'Gradient magnitude','png');
figure;
imshow(Angle,[])
title('Gradient angle')
saveas(gcf,'Gradient angle','png');
%%%%%%%%%%% define d %%%%%%%%%%%%%%%%
% x and x+pi are the same categories
a = Angle < 0;
```

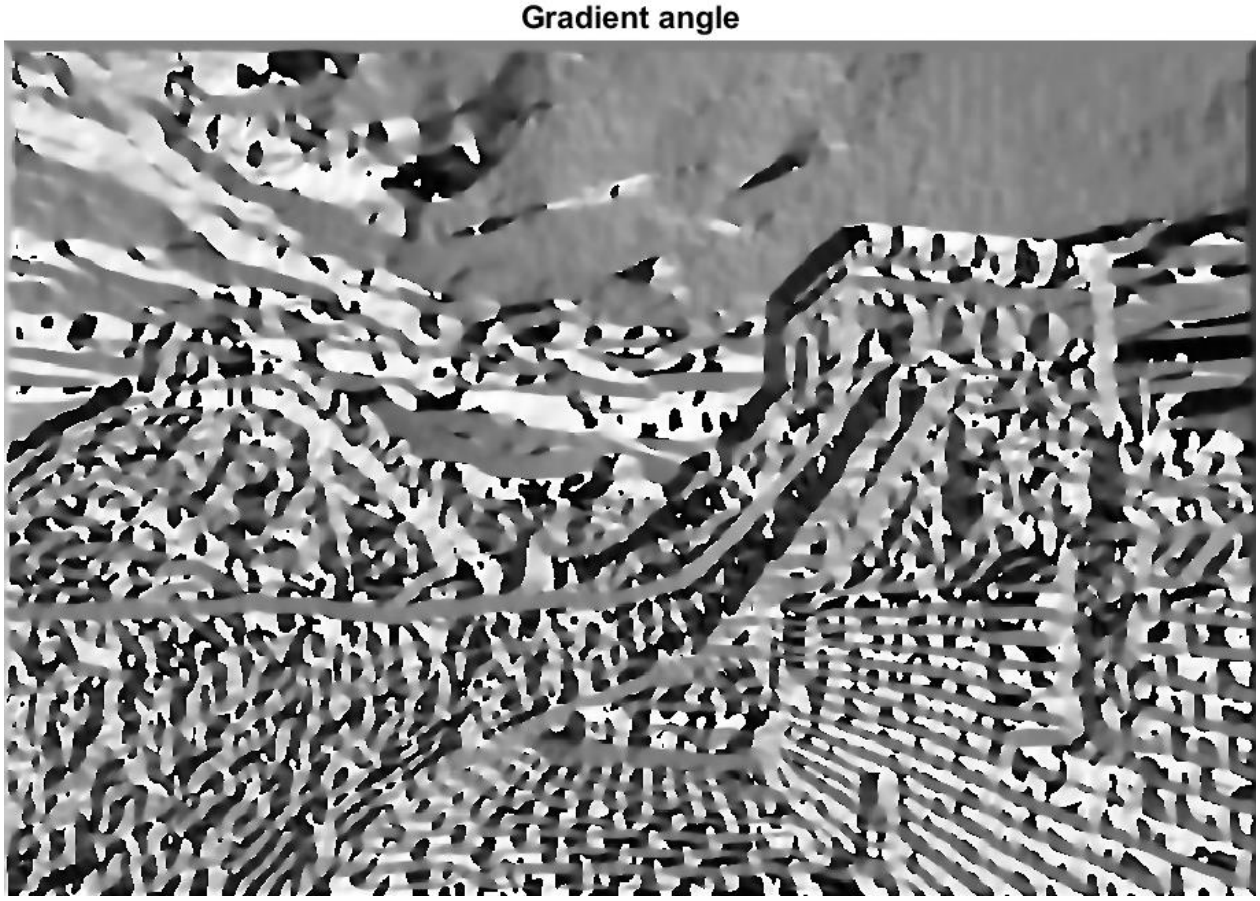
```
Angle = Angle + 180.0*a;
dmap = zeros(w,h);
d = [-inf 22.5 67.5 112.5 157.5 inf];
dmap((Angle >= d(1) & Angle < d(2)) | (Angle >= d(5) & Angle < d(6)))=1;
dmap(Angle >= d(2) & Angle < d(3))=2;
dmap(Angle >= d(3) & Angle < d(4))=3;
dmap(Angle >= d(4) & Angle < d(5))=4;

%%%%%%% compute gn %%%%%%%
gn = zeros(w,h);
temp = zeros(w+2,h+2);
temp(2:w+1,2:h+1)=dmap;
tempMag = zeros(w+2,h+2);
tempMag(2:w+1,2:h+1)=Mag;
for i = 2:w+1 %%%%%%%determine direction categories %%%%%%%
    for j = 2:h+1
        m = 0;
        n = 0;
        if temp(i,j)==1
            m = tempMag(i-1,j);
            n = tempMag(i+1,j);
        elseif temp(i,j)==2
            m = tempMag(i-1,j-1);
            n = tempMag(i+1,j+1);
        elseif temp(i,j)==3
            m = tempMag(i,j-1);
            n = tempMag(i,j+1);
        elseif temp(i,j)==4
            m = tempMag(i-1,j+1);
            n = tempMag(i+1,j-1);
        end

        if tempMag(i,j) >= max([n m])
            gn(i-1,j-1) = tempMag(i,j);
        else
            gn(i-1,j-1) = 0;
        end
    end
end
```

```
TH = 0.1;
TL = 0.04;
%%%%%%% determine gnh & gnl%%%%%%%
gnh = zeros(w,h);
gnl = zeros(w,h);
gnh(gn>=TH) = gn(gn>=TH);
gnl(TH > gn & gn >= TL) = gn(TH > gn & gn >= TL);
figure;
imshow(gnl,[])
title('gnL')
saveas(gcf,'gnL','png');
figure;
imshow(gnh,[])
title('gnH')
saveas(gcf,'gnH','png');
e =zeros(w,h);
temp_gnl=zeros(w+2,h+2);
temp_gnl(2:w+1,2:h+1)=gnl;
for i = 2:w+1
    for j = 2:h+1
        if gnh(i-1,j-1) > 0
            e(i-1,j-1)=1;
            if temp_gnl(i-1,j-1) > 0
                e(i-1-1,j-1-1)=1;
            end
            if temp_gnl(i-1,j) > 0
                e(i-1-1,j-1)=1;
            end
            if temp_gnl(i-1,j+1) > 0
                e(i-1-1,j+1-1)=1;
            end
            if temp_gnl(i,j-1) > 0
                e(i-1,j-1-1)=1;
            end
            if temp_gnl(i,j+1) > 0
                e(i-1,j+1-1)=1;
            end
            if temp_gnl(i+1,j-1) > 0
```

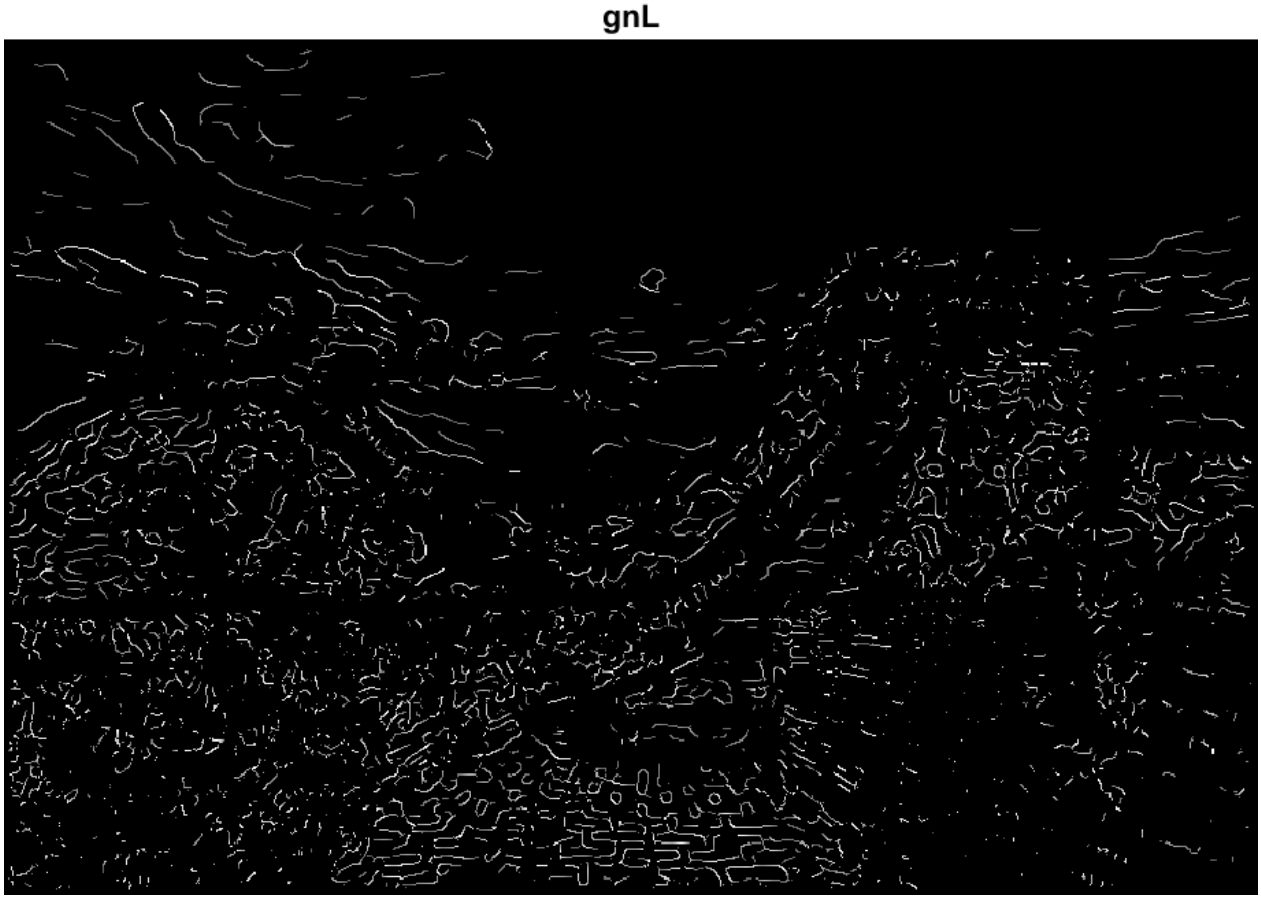
```
                    e(i+1-1,j-1-1)=1;
                end
                if temp_gnl(i+1,j) > 0
                    e(i+1-1,j-1)=1;
                end
                if temp_gnl(i+1,j+1) > 0
                    e(i+1-1,j+1-1)=1;
                end
            end
        end
    end
    figure;
    imshow(e)
    title('Final edge map')
    saveas(gcf,'Final edge map','png');
```

II. Figures of the gradient magnitude(normalized)and gradient angle (normalized)images





III. Figures of $g_{NL}(x, y)$ (normalized) and $g_{NH}(x, y)$ (normalized)



IV. Figure of final edge map