

Sample of Code

Xiaochen Wang

```
%*****edge_detection.m*****%
clear;
img = imread('head.jpg');           % read a picture
Edge = edgecolor(img);             % calling edgecolor function
subplot(1,2,1);
imshow(Edge);
imwrite(Edge,'head edge.bmp');

img1 = rgb2ycbcr(img);
img1 = img1(:, :, 1);

[wide,height] = size(img1);
img1 = double(img1);
Direct_map = zeros(wide,height);
mask_x = [-1,-2,-1;
          0, 0,0;
          1, 2,1];
mask_y = [-1,0,1;
          -2,0,2;
          -1,0,1];

for i = 2:wide-1
    for j = 2:height-1
        if(Edge(i,j) == 255)
            temp_x = img1(i-1:i+1,j-1:j+1).*mask_x;
            temp_y = img1(i-1:i+1,j-1:j+1).*mask_y;
            dx = sum(sum(temp_x));
            dy = sum(sum(temp_y));
            direct = angle(dx,dy);           % calling angle function
            Direct_map(i,j) = direct;
        end
    end
end

step1 = floor(wide/4);           % edge direction histogram
step2 = floor(height/4);
di = 1;
vector1 = zeros(9,9);
vector2 = zeros(1,81);

for i = 1:3
    for j = 1:3
        begin1 = 1+step1*(i-1);
        end1 = begin1+step1*2-1;
        begin2 = 1+step2*(j-1);
        end2 = begin2+step2*2-1;
        block = Direct_map(begin1:end1,begin2:end2);
```

```

vector = statistic(block); % calling statistic function
vector1(di,:) = vector;
begin = 1+(di-1)*9;
endd = begin + 8;
vector2(1,begin:endd) = vector;
di = di+1;

```

```

end
end

```

```

fid = fopen('outfile.txt','at'); % write data into text
fprintf(fid,'%3.5f ',vector2);
fprintf(fid,'%d\n',1);
fclose(fid);

```

```

%***** edge_detection.m *****%

```

```

%***** edgecolor.m *****%

```

```

function R = edgecolor(nm); % finding edges of a color image
img = nm;
[x y z] = size(img);
if z == 1
    rslt = edge(img,'canny'); % using Canny edge detector
elseif z == 3
    img1 = rgb2gray(img);
    dx1 = edge(img1,'canny',0.3);
    dx1 = (dx1*255);
end
R = dx1;

```

```

%***** edgecolor.m *****%

```

```

%***** angle.m *****%

```

```

function direct = angle(dx,dy) % edge direction

if (dx == 0)
    direct = 5;
else
    tan = dy/dx;
    if(tan >= -0.17632698 && tan < 0.17632698)
        direct = 1;
    elseif(tan >= 0.17632698 && tan < 0.57735027)
        direct = 2;
    elseif(tan >= 0.57735027 && tan < 1.23489716)
        direct = 3;
    elseif(tan >= 1.23489716 && tan < 2.74747742)
        direct = 4;
    elseif(tan >= 2.74747742);
        direct = 5;
    elseif(tan < -2.74747742)
        direct = 6;
    elseif(tan >= -2.74747742 && tan < -1.23489716)
        direct = 7;
    elseif(tan >= -1.23489716 && tan < -0.56735027)

```

```

        direct = 8;
    elseif(tan >= -0.57735027 && tan < -0.17632698)
        direct = 9;
    end
end
end
%***** angle.m *****%
%***** statistic.m *****%
function vector = statistic(block)                % accumulating edge direction

vector = zeros(1,9);
[wide,height] = size(block);
number = 0;
for i = 1:wide
    for j = 1:height
        if(block(i,j) ~= 0)
            vector(block(i,j)) = vector(block(i,j)) + 1;
            number = number + 1;
        end
    end
end
if(number ~= 0)
    vector = vector/number;
end
%***** statistic.m *****%

```

This code is used for edge detection. I use edge direction histogram (EDH) to character shape information of an object, and use 'Canny' edge detector to extract edge map. I divide evenly the bounding box of sub-pattern window into 4*4 grid, and accumulate a histogram of edge direction within the 2*2 grid.