Computer Vision I

Tim Luchterhand, Paul Nykiel (Group 17)

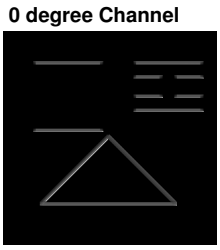
20. Juni 2018

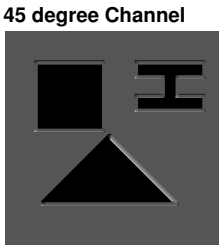
1 Moravec Operator

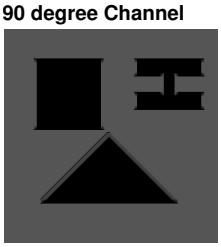
```
I = double(imread('figures1.png'))/255.0;

M1 = [0 -1 0;
      0 1 0;
      0 0 0];
M2 = [0 0 -1;
      0 1 0;
      0 0 0];

M3 = [0 0 0;
      0 1 -1;
      0 0 0];

M4 = [0 0 0;
      0 1 0;
      0 0 -1];

D1 = imfilter(I, M1,'replicate');
D2 = imfilter(I, M2,'replicate');
D3 = imfilter(I, M3,'replicate');
D4 = imfilter(I, M4,'replicate');

D1 = D1 .* D1;
D2 = D2 .* D2;
D3 = D3 .* D3;
D4 = D4 .* D4;

B = ones(5,5) * 1/25;

```

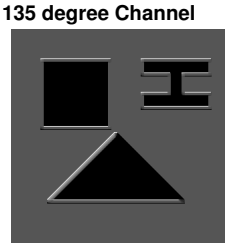
```
30 A1 = imfilter(D1, B,'replicate');
31 A2 = imfilter(D2, B,'replicate');
32 A3 = imfilter(D3, B,'replicate');
33 A4 = imfilter(D4, B,'replicate');
34
35 S = A1 + A2 + A3 + A4;
36
37 A1 = A1 ./ S;
38 A2 = A2 ./ S;
39 A3 = A3 ./ S;
40 A4 = A4 ./ S;
41
42 threshold = 0.15;
43 M1 = (A1 + A2 - abs(A1 - A2))/2; % Stolen from: https://
       stackoverflow.com/questions/26634232/element-wise-matrix-min
       -in-matlab
44 M2 = (A3 + A4 - abs(A3 - A4))/2;
45 M = (M1 + M2 - abs(M1- M2))/2;
46
47 underThreshold = find(M < threshold);
48 overThreshold = find(M >= threshold);
49 M(underThreshold) = 0;
50 M(overThreshold) = 1;
51
52 figure();
53
54 subplot(3,2,1);
55 imshow(A1);
56 title("0 degree Channel");
57 subplot(3,2,2);
58 imshow(A2);
59 title("45 degree Channel");
60 subplot(3,2,3);
61 imshow(A3);
62 title("90 degree Channel");
63 subplot(3,2,4);
64 imshow(A4);
65 title("135 degree Channel");
66 subplot(3,2,5);
67 imshow(M);
68 title("Detected Corners");
69 subplot(3,2,6);
70 imshow(I);
71 title("Original Image");
```

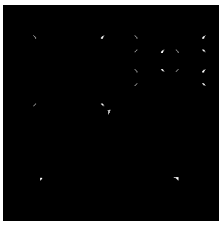
```
72
73 print("sh04ex01.eps", "-depsc");
```

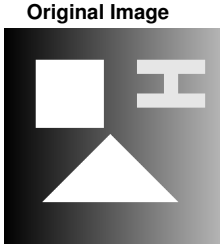












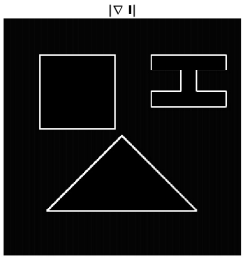
Uncorrelated noise has a high frequency and is therefor strongly amplified by the derivation. As a result there are a lot more false positives in the image. To achieve better results one could use a low-pass-filter to surpress high frequencys (in this case the noise). This can be done by convolving the image with a gaussian kernel.

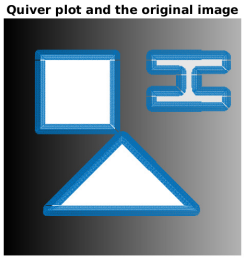
2 Structure Tensor

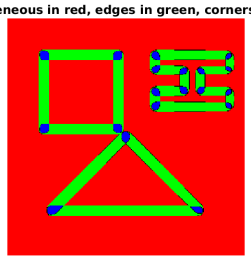
```
1 I = double(imread('figures1.png'))/255.0;
2
3 Sx = [-1  0  1;
4       -2  0  2;
5       -1  0  1];
6 Sy = [-1 -2 -1;
7        0  0  0;
8        1  2  1];
9
```

```
10 Ix = imfilter(I, Sx, 'replicate');
11 Iy = imfilter(I, Sy, 'replicate');
12
13 M = sqrt(Ix .* Ix + Iy .* Iy);
14
15 G = fspecial('gaussian',13,3);
16 S11 = imfilter(Ix .* Ix, G, 'conv');
17 S21 = imfilter(Iy .* Ix, G, 'conv');
18 S12 = imfilter(Ix .* Iy, G, 'conv');
19 S22 = imfilter(Iy .* Iy, G, 'conv');
20
21 x = zeros(size(Ix,1) * size(Ix,2), 1);
22 y = zeros(size(Ix,1) * size(Ix,2), 1);
23 u = zeros(size(Ix,1) * size(Ix,2), 1);
24 v = zeros(size(Ix,1) * size(Ix,2), 1);
25 eig1 = zeros(size(Ix,1) * size(Ix,2), 1);
26 eig2 = zeros(size(Ix,1) * size(Ix,2), 1);
27
28 H = zeros(size(I));
29 E = zeros(size(I));
30 C = zeros(size(I));
31
32 for yPos = 1:size(Ix,1)
33     for xPos = 1:size(Ix,2)
34         [V,D] = eig([S11(yPos,xPos) S12(yPos,xPos); S21(yPos,xPos
              ) S22(yPos,xPos)]);
35         x((yPos-1)*size(Ix,2) + xPos) = xPos;
36         y((yPos-1)*size(Ix,2) + xPos) = yPos;
37
38         e1 = max(D(1,1),D(2,2));
39         e2 = min(D(1,1),D(2,2));
40
41         eig1((yPos-1)*size(Ix,2) + xPos) = e1;
42         eig2((yPos-1)*size(Ix,2) + xPos) = e2;
43
44         if e1 > 0.03
45             u((yPos-1)*size(Ix,2) + xPos) = V(1,1);
46             v((yPos-1)*size(Ix,2) + xPos) = V(2,1);
47         else
48             u((yPos-1)*size(Ix,1) + xPos) = 0;
49             v((yPos-1)*size(Ix,1) + xPos) = 0;
50         end
51
52         if e1 <= 0.03 && e2 <= 0.03
```

```
53              H(yPos, xPos) = 1;
54          end
55
56          if e1 > 0.05 && e2 <= 0.03
57              E(yPos,xPos) = 1;
58          end
59
60          if e1 > 0.05 && e2 > 0.05
61              C(yPos, xPos) = 1;
62          end
63      end
64  end
65
66  figure();
67
68  subplot(1,3,1);
69  imshow(M);
70  title("|\nabla I|")
71
72  subplot(1,3,2);
73  imshow(I);
74  hold on;
75  quiver(x,y,u,v);
76  title("Quiver plot and the original image");
77
78  subplot(1,3,3);
79  imshow(cat(3,H,E,C));
80  title("Homogeneous in red, edges in green, corners in blue");
81
82  print("sh04ex02.eps", "-depsc");
```


|▽ I|


Quiver plot and the original image


Homogeneous in red, edges in green, corners in blue