

Computer Vision I

Tim Luchterhand, Paul Nykiel (Group 17)

9. Juli 2018

1 Gaussian Pyramids

```
1 boats = im2double(imread('boats.tif'));
2
3 G = cell(1,6);
4 S = cell(1,6);
5 G{1} = boats;
6 S{1} = boats;
7
8 figure;
9
10 subplot(2,6,1);
11 imshow(G{1});
12 title({"Subsampled by 1","with Gaussian Blur"});
13 subplot(2,6,7);
14 imshow(S{1});
15 title({"Subsampled by 1","without Gaussian Blur"});
16
17
18 for c=2:6
19     blurred = imgaussfilt(G{c-1});
20     G{c} = blurred(1:2:end, 1:2:end);
21     S{c} = S{c-1}(1:2:end, 1:2:end);
22     subplot(2,6,c);
23     imshow(G{c});
24     title({"Subsampled by " + 2^(c-1),"with Gaussian Blur"});
25     subplot(2,6,c+6);
26     imshow(S{c});
27     title({"Subsampled by " + 2^(c-1),"without Gaussian Blur"});
28 end
29
```

```
30 print("sh05ex01.eps", "-depsc");
```



Abbildung 1: Output of the matlab script

2 Laplacian Pyramids

2.1

```
1 % Part 1
2 boats = im2double(imread('boats.tif'));
3 G = cell(1,6);
4 G{1} = boats;
5 for c=2:6
6     blurred = imgaussfilt(G{c-1});
7     G{c} = blurred(1:2:end, 1:2:end);
8     S{c} = S{c-1}(1:2:end, 1:2:end);
9 end
10
11 Ge = cell(1,5);
12 for c=1:5
13     Ge{c} = imresize(G{c+1}, 2);
14 end
15
16 L = cell(1,6);
17 L{6} = G{6};
18 for c=1:5
19     L{c} = G{c} - Ge{c};
20 end
```

```

21
22 figure ;
23 for c=1:6
24     subplot(6, 3, (c-1)*3+1);
25     imshow(G{7-c});
26     title("G(" + (7-c) + ")");
27     if c > 1
28         subplot(6, 3, (c-1)*3+2);
29         imshow(Ge{7-c});
30         title("G_e(" + (7-c) + ")");
31     end
32     subplot(6, 3, (c-1)*3+3);
33     imshow(L{7-c});
34     title("L(" + (7-c) + ")");
35 end
36 print("sh05ex02_1.eps", "-depsc");

```

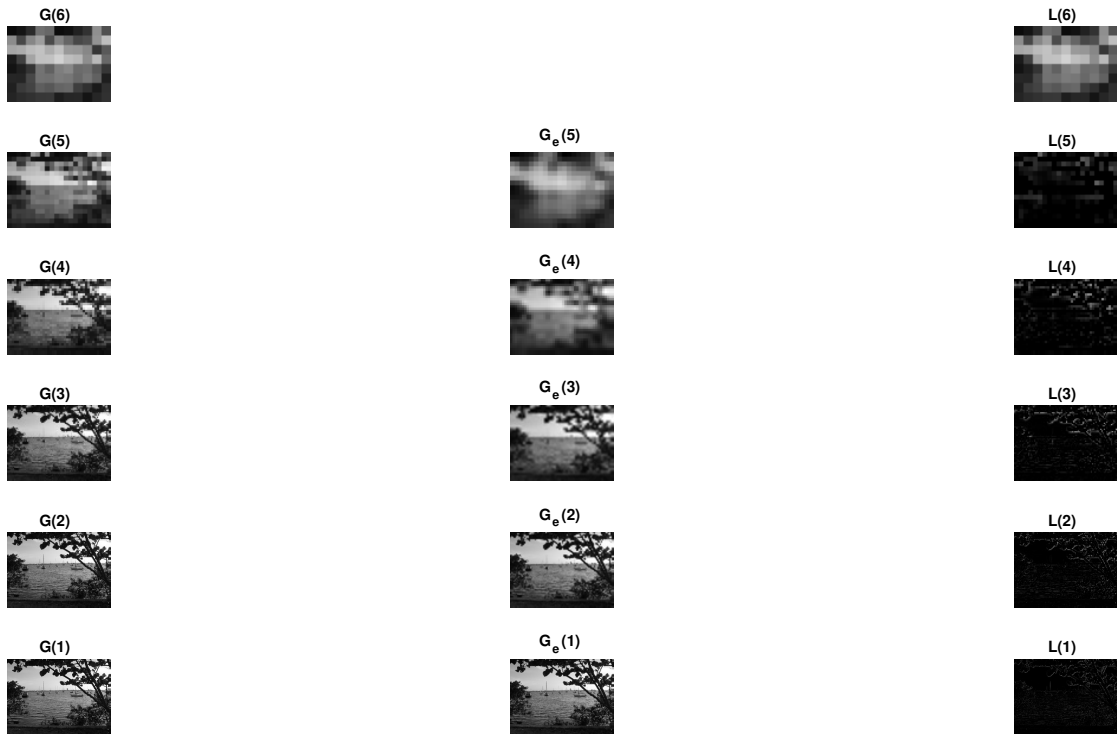


Abbildung 2: Output of the matlab script

2.2

```

38 % Part 2

```

```

39 E = cell(1, 5);
40 Gr = cell(1,5);
41 E{5} = imresize(L{6}, 2);
42 for c = 1:5
43     Gr{6-c} = E{6-c} + L{6-c};
44     if(c < 5)
45         E{5-c} = imresize(Gr{6-c}, 2);
46     end
47 end
48
49
50 figure;
51 for c=1:6
52     subplot(6, 3, (c-1)*3+1);
53     imshow(L{7-c});
54     title("L(" + (7-c) + ")");
55     if c > 1
56         subplot(6, 3, (c-1)*3+2);
57         imshow(E{7-c});
58         title("E(" + (7-c) + ")");
59         subplot(6, 3, (c-1)*3+3);
60         imshow(Gr{7-c});
61         title("G_r(" + (7-c) + ")");
62     end
63 end
64 print("sh05ex02_2.eps", "-depsc");

```

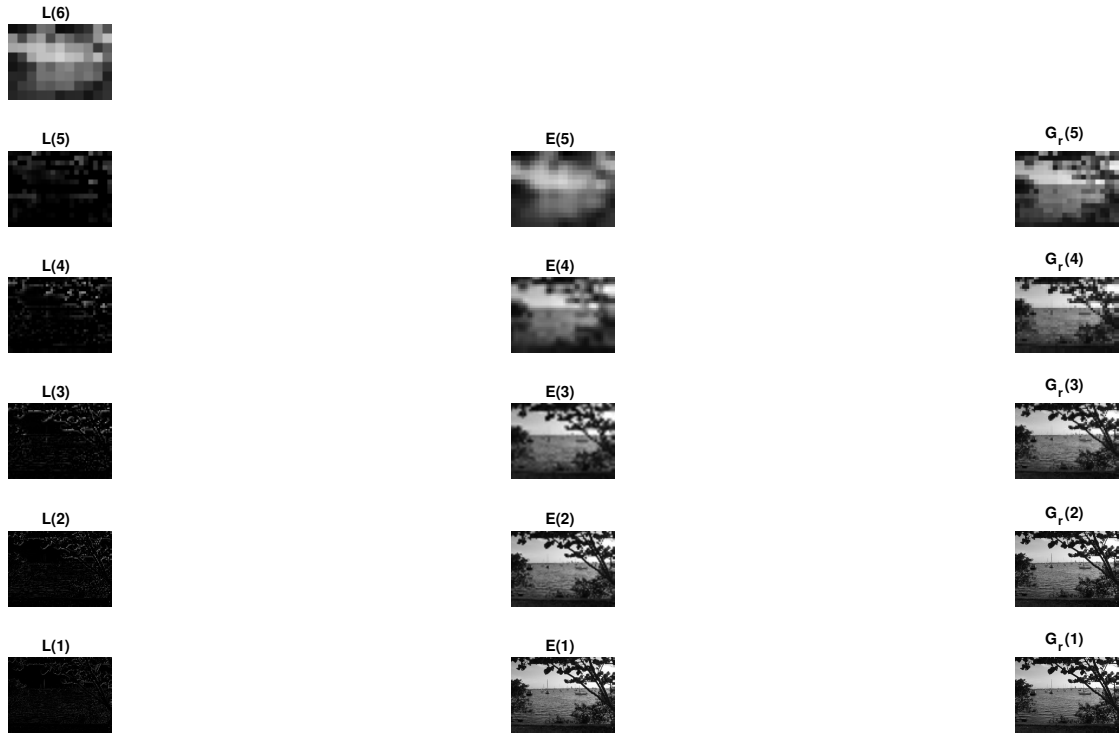


Abbildung 3: Output of the matlab script

2.3

```

66 % Part 3
67 L_thresh = cell(1,6);
68 L_thresh{6} = L{6};
69 for c = 1:5
70     m = max(max(L{c}));
71     t = m * 0.2;
72     L_thresh{c} = L{c};
73     L_thresh{c}(find(abs(L_thresh{c}) < t)) = 0;
74 end
75
76 E_reduced = cell(1, 5);
77 Gr_reduced = cell(1,5);
78 E_reduced{5} = imresize(L_thresh{6}, 2);
79 for c = 1:5
80     Gr_reduced{6-c} = E_reduced{6-c} + L_thresh{6-c};
81     if(c < 5)
82         E_reduced{5-c} = imresize(Gr_reduced{6-c}, 2);
83     end

```

```

84 end
85
86 figure;
87 for c=1:6
88     subplot(6, 3, (c-1)*3+1);
89     imshow(L{7-c});
90     title("L(" + (7-c) + ")");
91     if c > 1
92         subplot(6, 3, (c-1)*3+2);
93         imshow(E_reduced{7-c});
94         title("E(" + (7-c) + ") with Threshold");
95         subplot(6, 3, (c-1)*3+3);
96         imshow(Gr_reduced{7-c});
97         title("G_r(" + (7-c) + ") with Threshold");
98     end
99 end
100 print("sh05ex02_3_1.eps", "-depsc");
101
102 figure();
103 subplot(1,2,1);
104 imshow(Gr{1});
105 subplot(1,2,2);
106 imshow(Gr_reduced{1});
107 print("sh05ex02_3_2.eps", "-depsc");

```

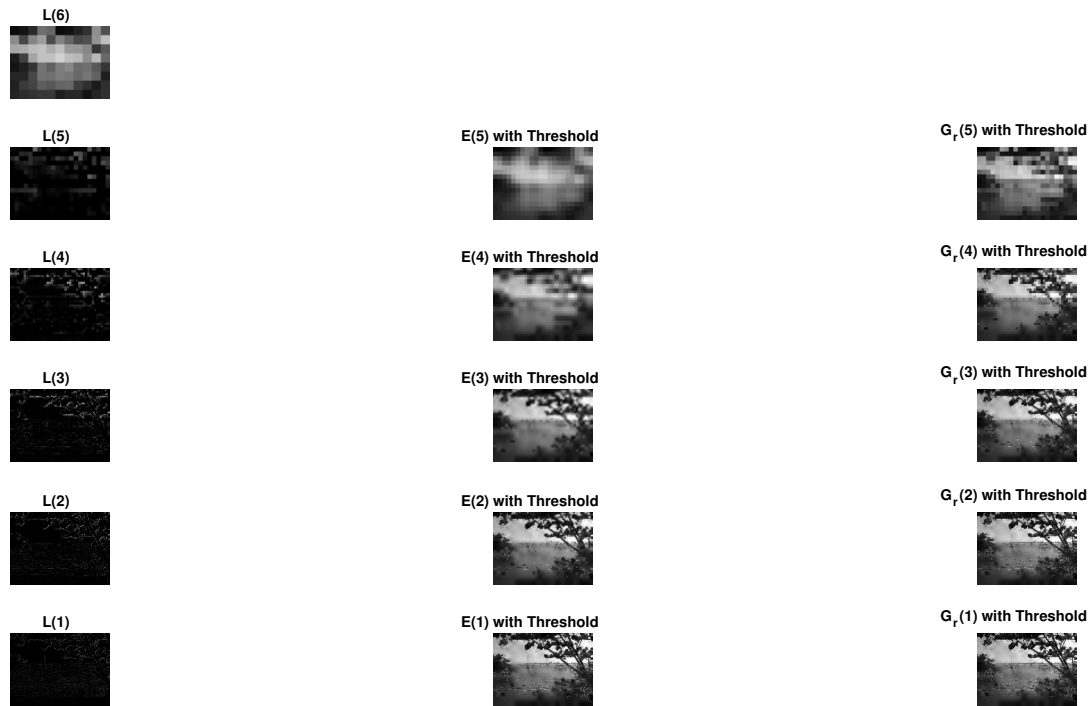


Abbildung 4: Output of the matlab script



Abbildung 5: Difference between the reconstructed and the original image

2.4

```

109 % Part 4
110 x = 0:0.05:0.3;
111 y = zeros(size(x));
112
113 for l=1:length(x)
114     L_thresh = cell(1,6);

```

```

115     L_thresh{6} = L{6};
116     for c = 1:5
117         m = max(max(L{c}));
118         t = m * x(1);
119         L_thresh{c} = L{c};
120         L_thresh{c}(find(abs(L_thresh{c}) < t)) = 0;
121     end
122
123     E_reduced = cell(1, 5);
124     Gr_reduced = cell(1,5);
125     E_reduced{5} = imresize(L_thresh{6}, 2);
126     for c = 1:5
127         Gr_reduced{6-c} = E_reduced{6-c} + L_thresh{6-c};
128         if(c < 5)
129             E_reduced{5-c} = imresize(Gr_reduced{6-c}, 2);
130         end
131     end
132     mse = sum(sum((G{1} - Gr_reduced{1}).^2)) / (size(G{1},1) *
133         size(G{1},2));
134     y(1) = mse;
135
136     end
137
138     figure;
139     plot(x, y);
140     xlabel("\lambda");
141     ylabel("MSE");
142     title("Mean square error as a function of the threshold");
143     print("sh05ex02_4.eps", "-depsc");

```

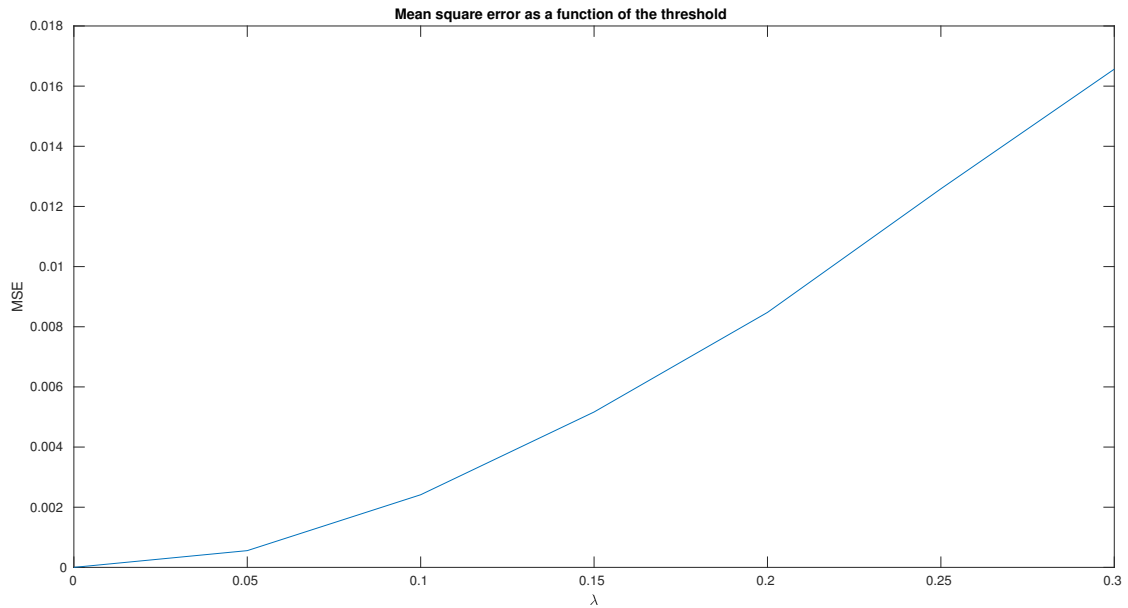



Abbildung 6: Output of the matlab script

3 Gabor Wavelets

```

1 % Part 1
2 basket = im2double(imread('basket.jpg'));
3 thetas = [0 pi/4 pi/2] * 180/pi;
4 fs = [0.64 0.32 0.08];
5
6 filters = cell(1,9);
7 b = 2.32;
8 x = zeros(1,9);
9 y = zeros(1,9);
10
11 for theta = 1:length(thetas)
12     for f = 1:length(fs)
13         sigma = b / fs(f);
14         filters{(theta-1)*3 + f} = getGabor(fs(f), sigma, thetas(
15             theta));
16         x((theta-1)*3 + f) = thetas(theta);
17         y((theta-1)*3 + f) = fs(f);
18     end
19 end
20 % Part 2

```

```

21 energy = zeros(1,9);
22 filtered = cell(1,9);
23
24 for c = 1:9
25     filtered{c} = imfilter(basket, filters{c}, 'conv');
26     figure;
27     imshow(abs(filtered{c}));
28     energy(c) = sqrt(sum(sum(abs(filtered{c}).^2)));
29 end
30
31 % Part 3
32 figure;
33 stem3(x, y, energy);
34 xlabel("\theta");
35 ylabel("f");
36 print("sh05ex03.eps", "-depsc");

```

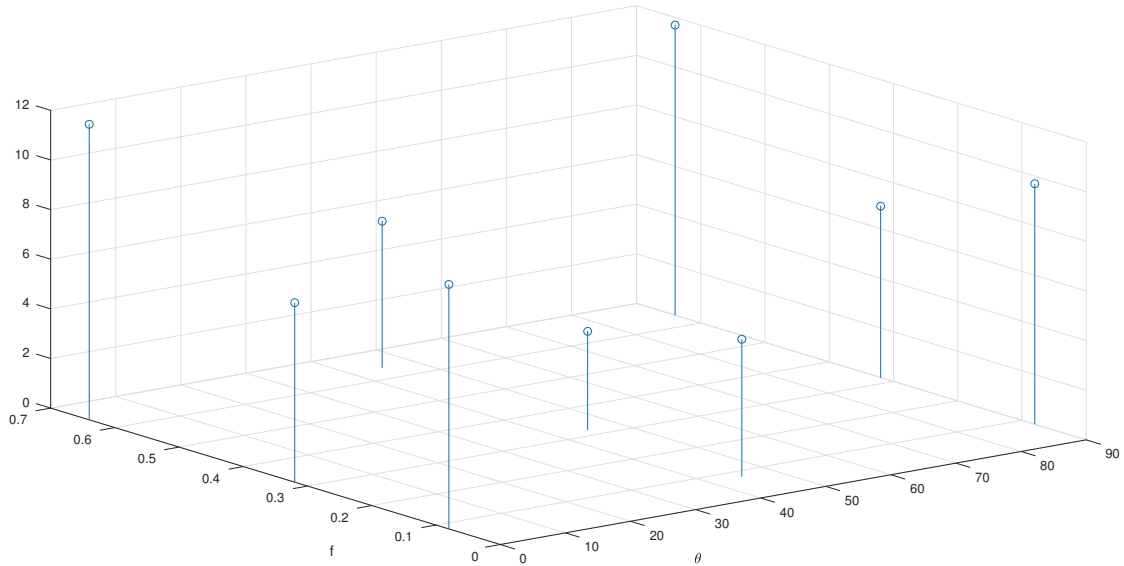


Abbildung 7: Output of the matlab script

For the orientation $\theta = 45^\circ$ all the frequency energies are lower compared to the other orientations since there are no significant image structures like edges.

In the other directions ($\theta = 0^\circ$ and $\theta = 90^\circ$) there are peaks at high frequencies due to prominent edges, as well as peaks at low frequencies as a result of partially homogeneous regions.