

```

32 % Revision: 1.0
33 % Date: 2007/05/09
34 % Author: Henrik Wolkesson
35
1 36 if ~IsScalar(sigma1) || sigma1 < 0 || ...
37     ~IsScalar(delta1) || delta1 < 0 || ...
38     ~IsScalar(sigma2) || sigma2 < 0 || ...
39     ~IsScalar(delta2) || delta2 < 0
40     error('Sigma1, Sigma2, Delta1 and Delta2 must be
41 end
1 42 if ~IsScalar(s2) || s2 < 0 || ~IsScalar(alpha) || alp
43     ~IsScalar(beta) || beta < 0 || ~IsScalar(tau)
44     error('s2, alpha, beta and tau must be a positive
45 end
1 46 if ~IsPositiveInteger(numIter)
47     error('numIter must be positive integer');
48 end
1 49 if useAOS < 0 || useAOS > 1 || round(useAOS) ~= useAC
50     error('useAOS must be either 0 or 1');
51 end
1 52 if graphs < 0 || graphs > 1 || round(graphs) ~= graph
53     error('Graphs must be either 0 or 1');
54 end
55
56 % Section 1: Initize
1 57 inImage = double(inImage);
1 58 I=inImage;
< 0.01 1 59 N = size(I,1);
< 0.01 1 60 M = size(I,2);
0.02 1 61 nablaI=zeros(N,M,2);
1 62 if (useAOS == 1)
63     unitMatrixM = sparse(eye(M));
64     unitMatrixN = sparse(eye(N));
65 end
66
67 % Section 2: Create kernels
1 68 derivKernel = createDerivKernel(sigma1, delta1);
1 69 Kx = createGaussKernel(sigma2,delta2);
70
1 71 if (graphs == 1)
72     figure('Name', 'Kernels');
73     subplot(1,2,1);plot(Kx);
74     title(['Gauss Kernel (Size: ' num2str(size(Kx,2))
75     subplot(1,2,2);plot(derivKernel);
76     title(['Derivating Kernel (Size: ' num2str(size(d
77 end;
78
1 79 if (size(Kx,2)>size(I,2))
80     warning('Derivating kernel is bigger than image')
81 end
82
< 0.01 1 83 tic;
1 84 for k = 1:numIter
85     % Section 3: Calculate derivatives
0.03 2 86 I_y=imfilter(I,derivKernel', 'conv', 'replicate')
0.02 2 87 I_x=imfilter(I,derivKernel, 'conv', 'replicate');
88

```