Contents

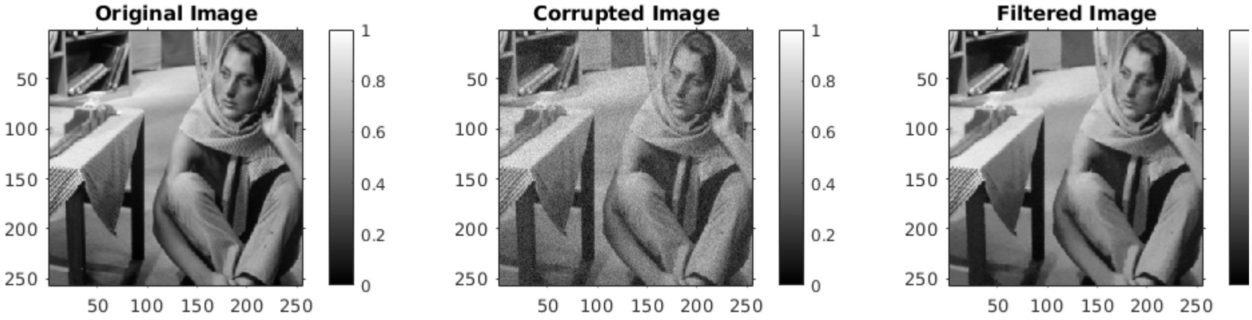
MyMainScript

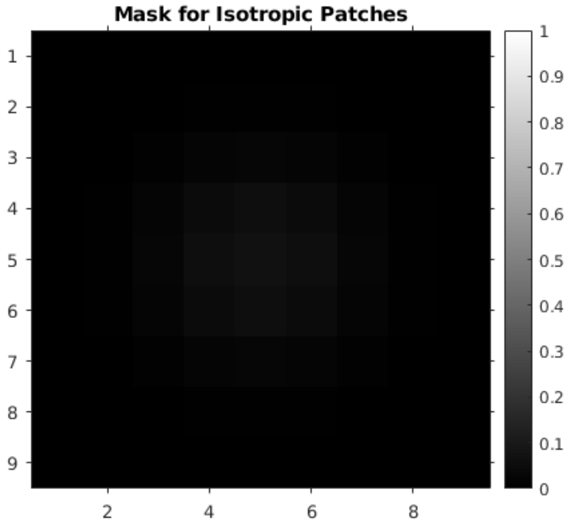
```
tic;
```

Filtering Barbara image

```
load('../data/barbara.mat');
imageOrig = imgaussfilt(imageOrig,0.66);
imageOrig = imresize(imageOrig,0.5);
[len, wid] = size(imageOrig);
imgCorrupt = imageOrig + 0.05*(max(max(imageOrig))-min(min(imageOrig)))*randn(len);

h = 1.26;
patchSize = [9,9];
windowSize = [25,25];
out = myPatchBasedFiltering(imgCorrupt, patchSize, windowSize, h);
rsmd0 = myRsmd(out,imageOrig);
display(rsmd0);
myDisplayThreeImage(imageOrig,imgCorrupt,out);

out1 = myPatchBasedFiltering(imgCorrupt, patchSize, windowSize, h*0.9);
rsmd1 = myRsmd(out1,imageOrig);
display(rsmd1);

out2 = myPatchBasedFiltering(imgCorrupt, patchSize, windowSize, h*1.1);
rsmd2 = myRsmd(out2,imageOrig);
display(rsmd2);

figure;
gaussianFilter = fspecial('gaussian', patchSize, double(patchSize(1))/6);
imshow(gaussianFilter,'InitialMagnification','fit');
title("Mask for Isotropic Patches")
colormap(gray(200));
axis on;
colorbar;
```

```
optimal h-> rsmd0 = 2.4216 (h=1.26)

0.9*h-> rsmd1 = 2.4295 (h=1.26*0.9)

1.1*h-> rsmd2 = 2.4309 (h=1.26*1.1)
```

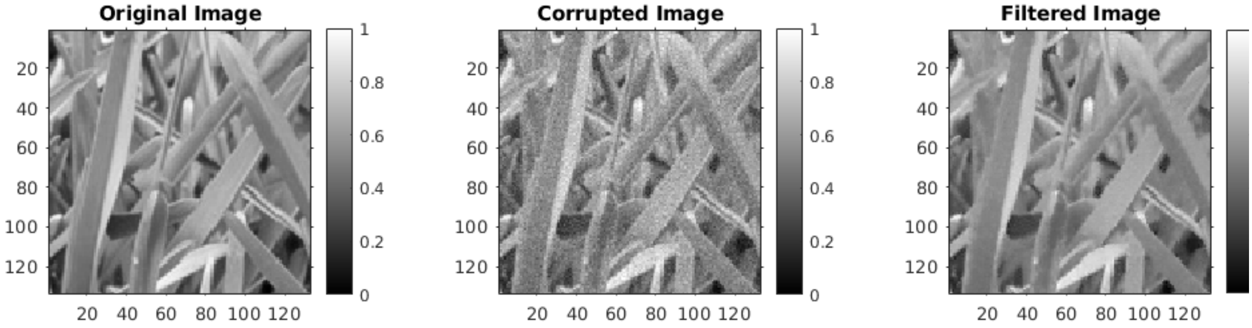


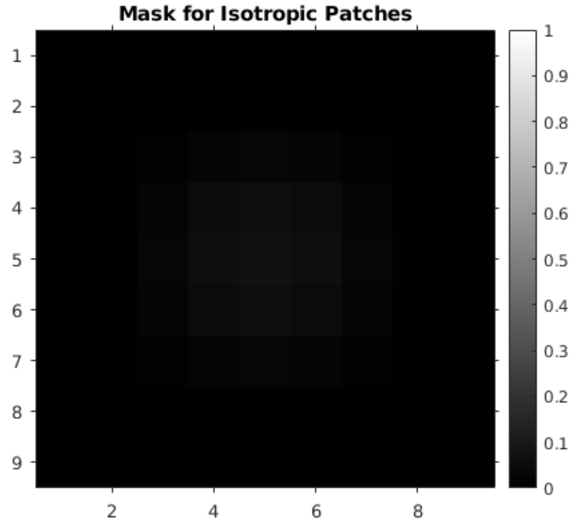


Filtering Grass image

```
imageOrig = im2double(imread('../data/grass.png'));
[len, wid] = size(imageOrig);
imgCorrupt = imageOrig + 0.05*(max(max(imageOrig))-min(min(imageOrig)))*randn(len);

h = 0.000109;
patchSize = [9,9];
windowSize = [25,25];
out = myPatchBasedFiltering(imgCorrupt, patchSize, windowSize, h);
rsmd0 = myRsmd(out,imageOrig);
display(rsmd0);
myDisplayThreeImage(imageOrig,imgCorrupt,out);

out1 = myPatchBasedFiltering(imgCorrupt, patchSize, windowSize, h*0.9);
rsmd1 = myRsmd(out1,imageOrig);
display(rsmd1);

out2 = myPatchBasedFiltering(imgCorrupt, patchSize, windowSize, h*1.1);
rsmd2 = myRsmd(out2,imageOrig);
display(rsmd2);

figure;
gaussianFilter = fspecial('gaussian', patchSize, double(patchSize(1))/6);
imshow(gaussianFilter,'InitialMagnification','fit');
title("Mask for Isotropic Patches")
colormap(gray(200));
axis on;
colorbar;
```

```
optimal h-> rsmd0 = 0.0291 (h=0.000109)

0.9*h-> rsmd1 = 0.0293 (h=0.000109*0.9)

1.1*h-> rsmd2 = 0.0293 (h=0.000109*1.1)
```



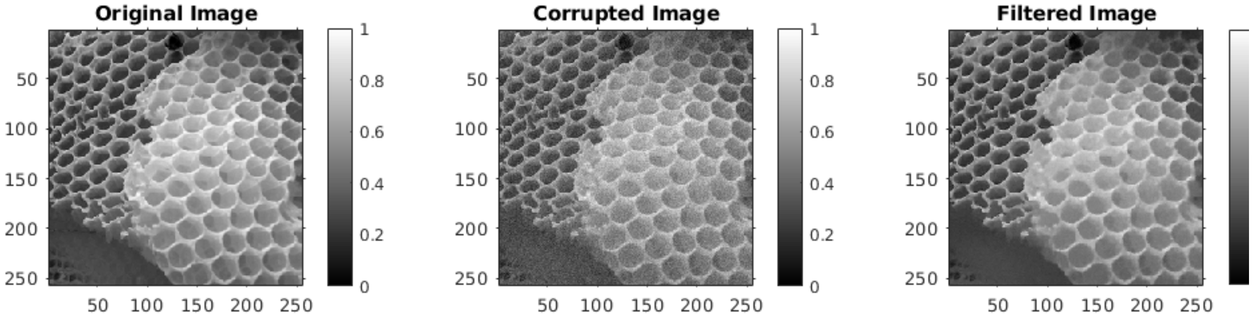


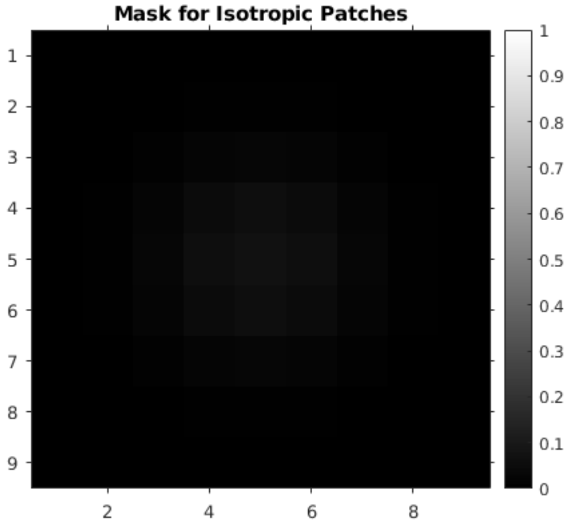
Filtering Honey Comb image

```
imageOrig = im2double(imread('../data/honeyCombReal.png'));
[len, wid] = size(imageOrig);
imgCorrupt = imageOrig + 0.05*(max(max(imageOrig))-min(min(imageOrig)))*randn(len);

h = 0.00013;
patchSize = [9,9];
windowSize = [25,25];
out = myPatchBasedFiltering(imgCorrupt, patchSize, windowSize, h);
rsmd0 = myRsmd(out,imageOrig);
display(rsmd0);
myDisplayThreeImage(imageOrig,imgCorrupt,out);

out1 = myPatchBasedFiltering(imgCorrupt, patchSize, windowSize, h*0.9);
rsmd1 = myRsmd(out1,imageOrig);
display(rsmd1);

out2 = myPatchBasedFiltering(imgCorrupt, patchSize, windowSize, h*1.1);
rsmd2 = myRsmd(out2,imageOrig);
display(rsmd2);
```

```
figure;
gaussianFilter = fspecial('gaussian', patchSize, double(patchSize(1))/6);
imshow(gaussianFilter,'InitialMagnification','fit');
title("Mask for Isotropic Patches")
colormap(gray(200));
axis on;
colorbar;

toc;
```

```
optimal h-> rsmd0 = 0.0291 (h=0.00013)

0.9*h-> rsmd1 = 0.0293 (h=0.00013*0.9)

1.1*h-> rsmd2 = 0.0291 (h=0.00013*1.1)

Elapsed time is 115.620351 seconds.
```





Published with MATLAB® R2018a