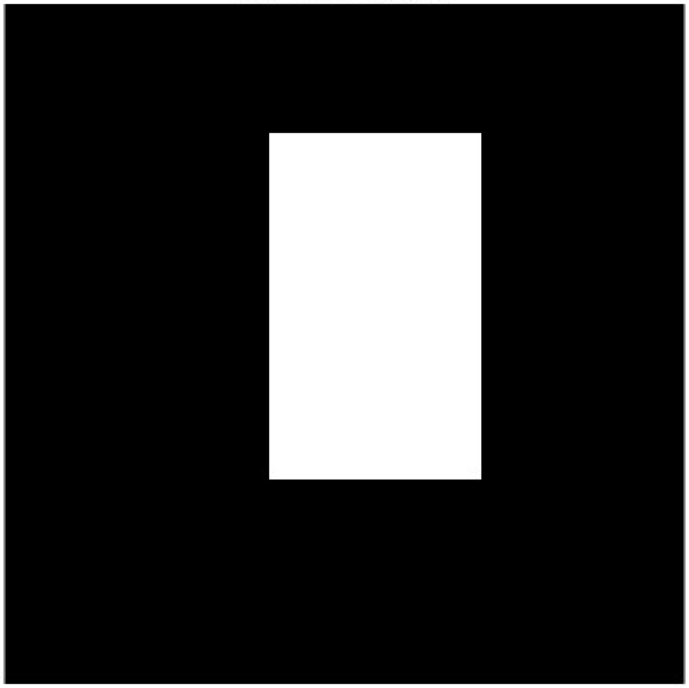
```
%==========================
%  Name:                hw7_2.m
%
%  Author:              Kairi Kozuma
%==========================

% 1) Diffuse for 2 images (1 blob, 1 westin)

% blob
IMAGE = double(blurme);
smoothI = isodiffuse(IMAGE, .1, 5000);

fh1 = figure(1);
  clf;
  imagesc(IMAGE);
  colormap('gray');
  axis image;
  axis off;
  title('Blob before diffusion');

fh2 = figure(2);
  clf;
  imagesc(smoothI);
  colormap('gray');
  axis image;
  axis off;
  title('Blob after diffusion');

% westin
IMAGE = westin;
smoothI = isodiffuse(IMAGE, .1, 600);

fh3 = figure(3);
  clf;
  imagesc(IMAGE);
  colormap('gray');
  axis image;
  axis off;
  title('westin before diffusion');

fh4 = figure(4);
  clf;
  imagesc(smoothI);
  colormap('gray');
  axis image;
  axis off;
  title('westin after diffusion');

% 2)

% Show Gaussian blur with increasing standard deviations
```

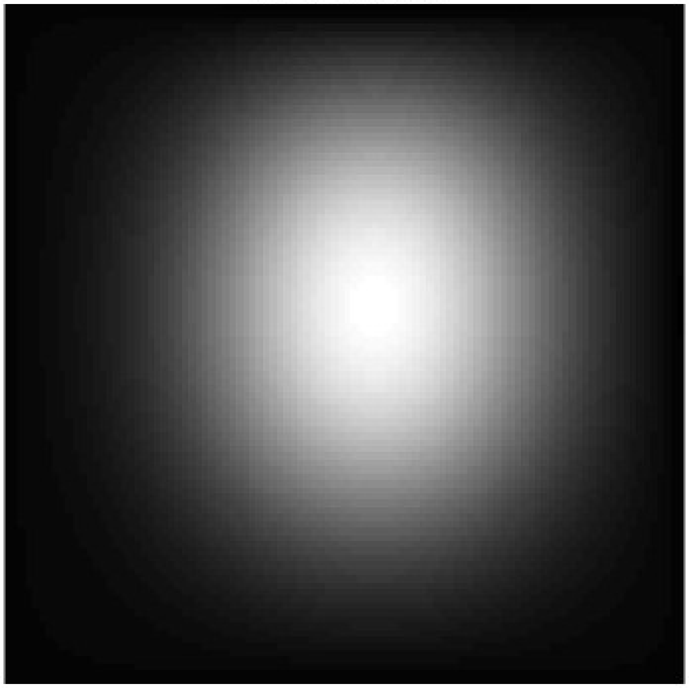
```matlab
t = [0.25, 0.5, 0.75, 1, 1.5, 2];
gaus = gaussianFilt(double(blurme), 2);
err = zeros(1,length(t));
for i = 1:length(t)
    diff = isodiffuse(double(blurme), .001, t(i) * 1000);
    err(i) = sumsqr(gaus - diff);
end
fh5 = figure(5);
plot(t, err);
title('Error (sumsqr) vs time');


% 3)
tarr = 2.^[0:10];
avg = zeros(1,length(tarr));
fh6 = figure(6);
for i=1:length(tarr)
    smoothI3 = isodiffuse(double(rcells), 0.01, tarr(i));
    avg(i) = mean(mean(smoothI3));
end

plot(tarr, avg);
title('Average vs time');
%=============================== isodiffuse ===============================
%
%  Performs isotropic diffusion by running the heat equation on a grayscale
%  image.
%
%  function sI = isodiffuse(gI, dt, iter)
%
%
%  Inputs:
%    gI          - grayscale/scalar image (should be floating point format).
%    dt          - timestep to use in running the heat equation
%                      the argument should be 0 < dt < 0.5.
%    iter        - total number of iterations.
%
%  Output:
%    sI      - the smoothed image.
%
%=============================== isodiffuse ===============================

%
%  Name:                 isodiffuse.m
%
%  Author:               Patricio A. Vela, pvela@gatech.edu
%
%  Created:              2010/01/05
%  Modified:    2010/01/05
%
%=============================== isodiffuse ===============================
function sI = isodiffuse(gI, dt, iter)
```

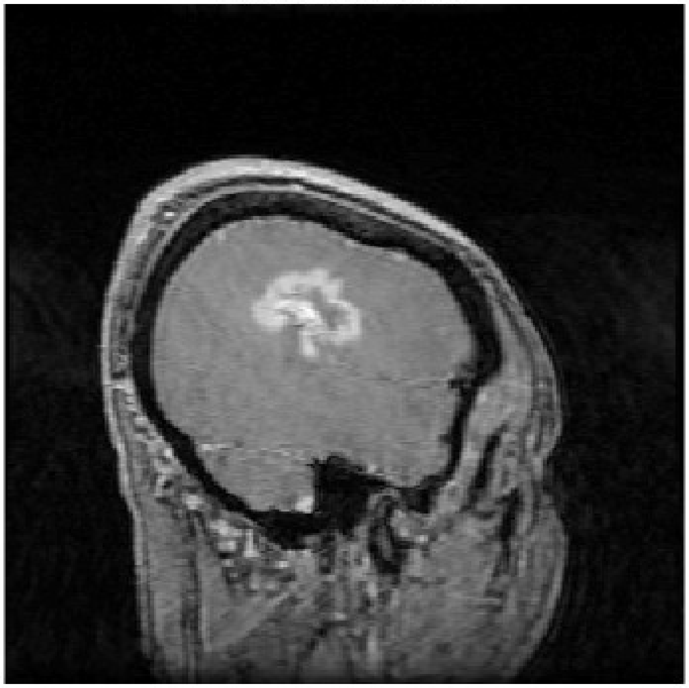
```
for i=1:iter
    dI = del2(gI);
    gI = gI + dI * dt;
end

sI = gI;

end
%
%=============================== isodiffuse ==============================


%============================ gaussianFilt ==============================
%
%  function [image] = gaussianFilt(image, sd)
%
%
%  INPUT:
%    image  - matrix representing image
%    sd     - standard deviation
%
%============================ gaussianFilt ==============================
function [image] = gaussianFilt(image, sd)

% Gaussian filter to use
sz = ceil(2.5 * sd);
if mod(sz,2) == 0
    sz = sz + 1;
end
gausfilt = fspecial('gaussian', [sz,sz], sd);

% Filter using the Gaussian blur
image = imfilter(image, gausfilt);

end
```

Blob before diffusion

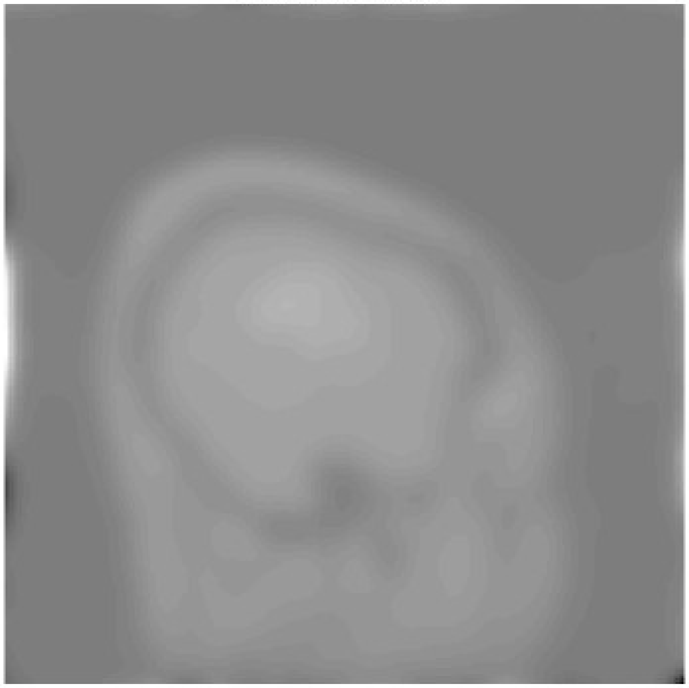


Blob after diffusion

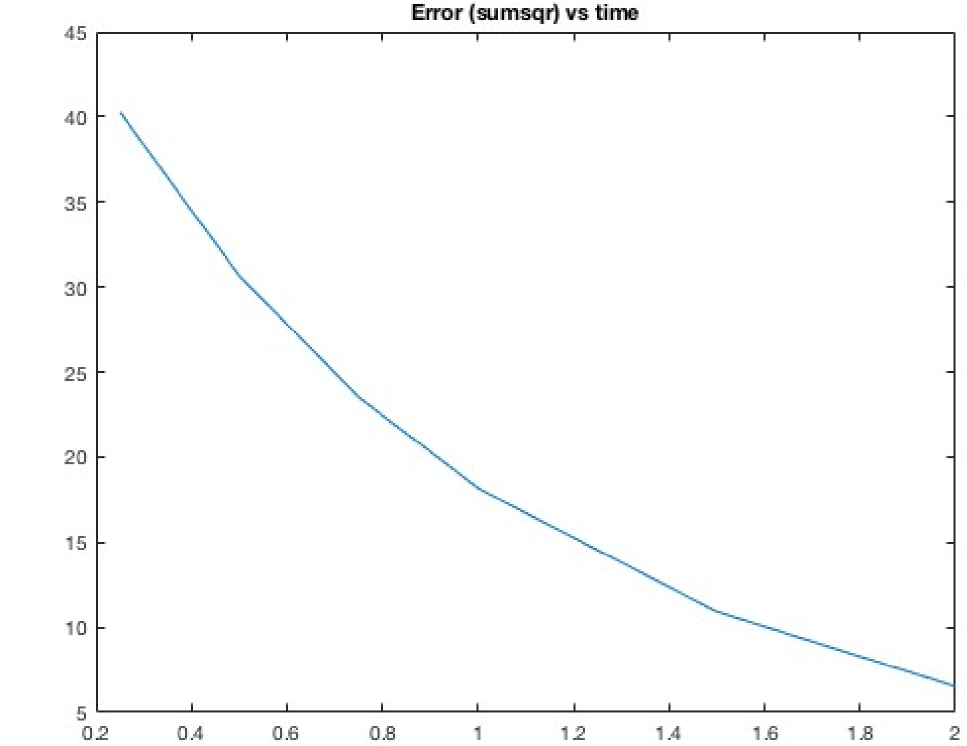




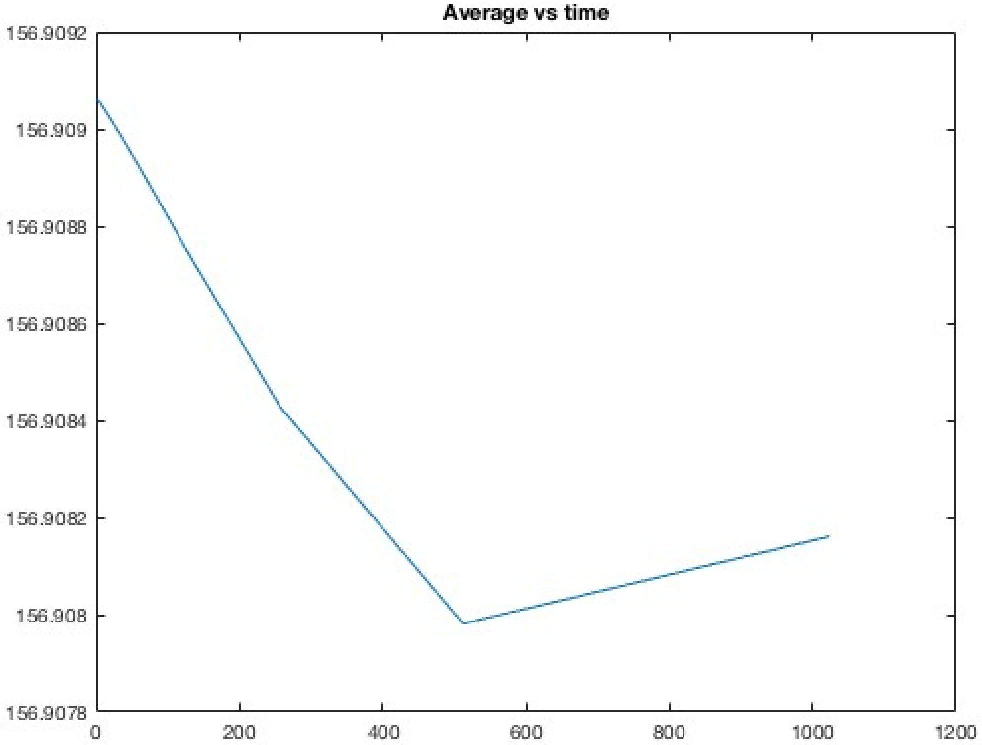
westin before diffusion



westin after diffusion



Error (sumsqr) vs time



Average vs time

Published with MATLAB® R2016b