Assignment A1: Basic Image Processing Functions

CS 4640
Spring 2018

Assigned: 9 January 2018

Due: 25 January 2018

For this problem, handin Matlab .m files for the functions described by the headers below. Note that one of these is a driver which creates inputs for each function and runs the function on those inputs to obtain the output.

None of the functions should write to the interpreter, draw, etc.

```
function im = CS4640_create_im(f_name,M,N,Q,x_min,x_max,y_min,y_max)
% CS4640_create_im - create an image from a function
% On input:
%     f_name (string): name of function
%     M (int): number of rows in image
%     N (int): number of cols in image
%     Q (int): number of quantization levels in image
%     x_min (float): min x value for planar patch
%     x_max (float): max x value for planar patch
%     y_min (float): min y value for planar patch
%     y_max (float): max y value for planar patch
% On output:
%     im (MxN array): image
% Call:
%     im = CS4640_create_im('CS4640_hemisphere',100,100,64,-4,4,-4,4);
% Author:
%     <Your Name>
%     UU
%     Spring 2018
```

```
%


function imd = CS4640_im_dist(im1,im2)
% CS4640_im_dist - pixel-wise vector distance between images
% On input:
%     im1 (M1xN1xP1 array): image 1
%     im2 (M2xN2xP2 array): image 2
% On output:
%     imd (M1xN1 array): distace image
% Call:
%     imd = CS4640_im_dist(im1,im2);
% Author:
%     <Your Name>
%     UU
%     Spring 2018
%


function im_FT = CS4640_FT(im)
% CS4640_FT - compute Fourier Transform of image
% On input:
%     im (MxN float array): input image
% On output:
%     im_FT (MxN float array): Fourier Transform of im
% Call:
%     im_FT = CS4640_FT(im);
% Author:
%     <Your name>
%     UU
%     Spring 2018
%


function im_FTi = CS4640_FTi(im)
% CS4640_FTi - compute inverse Fourier Transform of image
% On input:
%     im (MxN array): input image
% On output:
%     im_FTi (MxN float array): inverse Fourier Transform of im
% Call:
```

```
%     im_FTi = CS4640_FTi(im);
% Author:
%     <Your name>
%     UU
%     Spring 2018
%


function imr = CS4640_register(ref,im,pts)
% CS4640_register - register an image to a reference
% On input:
%     ref (M1xN1 array): reference image
%     im (M2xN2 array): input image
%     pts (nx4 array): corresponding pixels in the two images
%       in order x_ref y_ref v_im w_im
% On output:
%     imr (MxN array): registered version of im
% Call:
%     imr = CS4640_register(ref,im,pts);
% Author:
%     <Your name>
%     UU
%     Spring 2018
%


function imn =
CS4640_add_noise(im,p_min,p_max,noise_type,a,b,mu,sigma2)
% CS4640_add_noise - add uniform or Gaussian noise
% On input:
%     im (MxN image): input image
%     p_min (float): minimum gray level in noise image
%     p_max (float): maximum gray level in noise image
%     noise_type (int): 1: uniform; 2: Gaussian
%     a (float): lower limit on uniform range
%     b (float): upper limit on uniform range
%     mu (float): mean of N(mu,sigma2)
%     sigma2 (float): variance of N(mu,sigma2)
% On output
%     imn (MxN array): noise image
% Call:
```

```
%     imn = CS4640_add_noise(im3,2,10,1,0,1,0,.0001);
% Author:
%     <Your name>
%     UU
%     Spring 2018
%


function CS4640_A1_driver
% CS4640_A1_driver - driver for A1 functions
% On input:
%     N/A
% On output:
%     N/A
% Call:
%     CS4640_A1_driver
% Author:
%     <Your name>
%     UU
%     Spring 2018
%
```