Colin Keenan

ECE 5470

2-24-2020

Homework 4

1. Laplacian Enhancement

# Write program to implement the Laplacian enhancement technique described in Lecture Note 4. Use the mask shown in Slide 4-63 (c and d) on Fig-1a.jpg

clear

fprintf("Question 1, Part A:")

file = 'Fig4-1a.jpg';

image = imread(file);

image\_double = im2double(image);

image\_mod\_3x3\_hp = zeros(size(image\_double));

image\_mod\_3x3\_hp\_inv = zeros(size(image\_double));

mask\_3x3\_hp = [[-1 -1 -1]

[-1 8 -1]

[-1 -1 -1]];

for i = 1+1 : size(image\_double,1)-1

for j = 1+1 : size(image\_double,2)-1

frame\_3x3 = [[image\_double(i-1,j-1) image\_double(i-1,j+0) image\_double(i-1,j+1)]

[image\_double(i+0,j-1) image\_double(i+0,j+0) image\_double(i+0,j+1)]

[image\_double(i+1,j-1) image\_double(i+1,j+0) image\_double(i+1,j+1)]];

masked = frame\_3x3 .\* mask\_3x3\_hp;

image\_mod\_3x3\_hp(i,j) = sum(masked,"all");

end

end

mask\_3x3\_hp\_inv = [[1 1 1]

[1 -8 1]

[1 1 1]];

for i = 1+1 : size(image\_double,1)-1

for j = 1+1 : size(image\_double,2)-1

frame\_3x3 = [[image\_double(i-1,j-1) image\_double(i-1,j+0) image\_double(i-1,j+1)]

[image\_double(i+0,j-1) image\_double(i+0,j+0) image\_double(i+0,j+1)]

[image\_double(i+1,j-1) image\_double(i+1,j+0) image\_double(i+1,j+1)]];

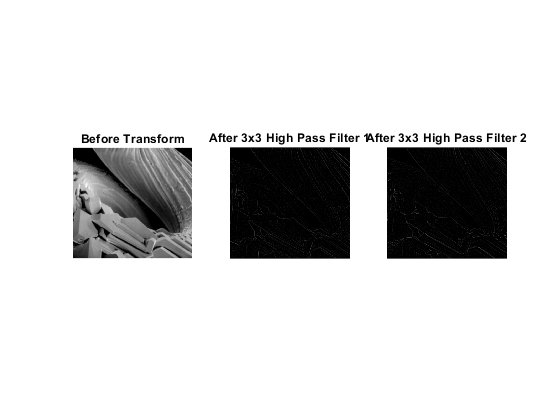
masked = frame\_3x3 .\* mask\_3x3\_hp\_inv;

image\_mod\_3x3\_hp\_inv(i,j) = sum(masked,"all");

end

end

figure();

subplot(1,3,1)

imshow(image);

title("Before Transform");

subplot(1,3,2)

imshow(image\_mod\_3x3\_hp);

title("After 3x3 High Pass Filter 1");

subplot(1,3,3)

imshow(image\_mod\_3x3\_hp\_inv);

title("After 3x3 High Pass Filter 2");

# Write program to implement the Laplacian enhancement technique described in Lecture Note 4. Use the mask shown in Slide 4-63 (c and d) on Fig-1b.jpg

clear

fprintf("Question 1, Part B:")

file = 'Fig4-1b.jpg';

image = imread(file);

image\_double = im2double(image);

image\_mod\_3x3\_hp = zeros(size(image\_double));

image\_mod\_3x3\_hp\_inv = zeros(size(image\_double));

mask\_3x3\_hp = [[-1 -1 -1]

[-1 8 -1]

[-1 -1 -1]];

for i = 1+1 : size(image\_double,1)-1

for j = 1+1 : size(image\_double,2)-1

frame\_3x3 = [[image\_double(i-1,j-1) image\_double(i-1,j+0) image\_double(i-1,j+1)]

[image\_double(i+0,j-1) image\_double(i+0,j+0) image\_double(i+0,j+1)]

[image\_double(i+1,j-1) image\_double(i+1,j+0) image\_double(i+1,j+1)]];

masked = frame\_3x3 .\* mask\_3x3\_hp;

image\_mod\_3x3\_hp(i,j) = sum(masked,"all");

end

end

mask\_3x3\_hp\_inv = [[1 1 1]

[1 -8 1]

[1 1 1]];

for i = 1+1 : size(image\_double,1)-1

for j = 1+1 : size(image\_double,2)-1

frame\_3x3 = [[image\_double(i-1,j-1) image\_double(i-1,j+0) image\_double(i-1,j+1)]

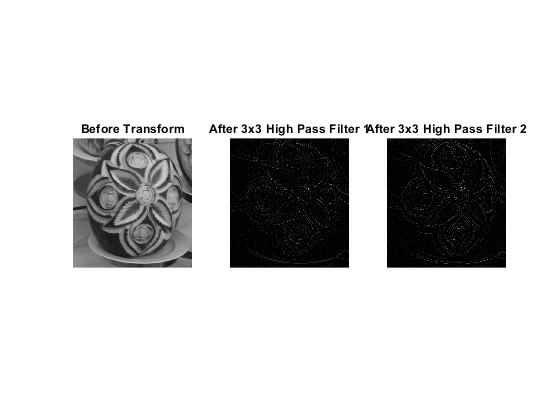
[image\_double(i+0,j-1) image\_double(i+0,j+0) image\_double(i+0,j+1)]

[image\_double(i+1,j-1) image\_double(i+1,j+0) image\_double(i+1,j+1)]];

masked = frame\_3x3 .\* mask\_3x3\_hp\_inv;

image\_mod\_3x3\_hp\_inv(i,j) = sum(masked,"all");

end

end

figure();

subplot(1,3,1)

imshow(image);

title("Before Transform");

subplot(1,3,2)

imshow(image\_mod\_3x3\_hp);

title("After 3x3 High Pass Filter 1");

subplot(1,3,3)

imshow(image\_mod\_3x3\_hp\_inv);

title("After 3x3 High Pass Filter 2");

2. Unsharpmasking and High boost Filtering

# Perform on Fig4-3.tif

clear

fprintf("Question 2:")

file = 'Fig4-2.tif';

image = imread(file);

image\_double = im2double(image);

image\_mod\_3x3\_smooth = zeros(size(image\_double));

mask\_3x3 = [[1 1 1]

[1 1 1]

[1 1 1]];

weight\_3x3 = sum(sum(mask\_3x3));

for i = 1+1 : size(image\_double,1)-1

for j = 1+1 : size(image\_double,2)-1

frame\_3x3 = [[image\_double(i-1,j-1) image\_double(i-1,j+0) image\_double(i-1,j+1)]

[image\_double(i+0,j-1) image\_double(i+0,j+0) image\_double(i+0,j+1)]

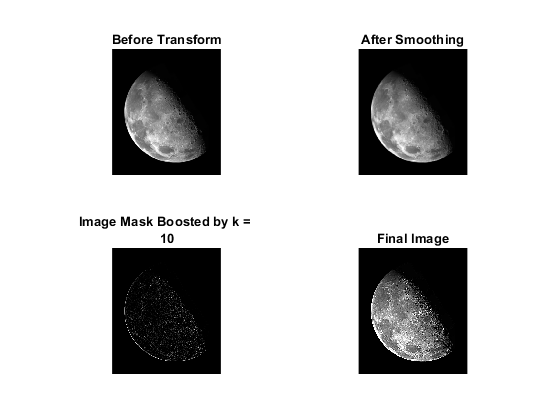
[image\_double(i+1,j-1) image\_double(i+1,j+0) image\_double(i+1,j+1)]];

masked\_3x3 = frame\_3x3 .\* mask\_3x3;

new\_level = sum(sum(masked\_3x3)) / weight\_3x3;

image\_mod\_3x3\_smooth(i,j) = new\_level;

end

end

image\_mask = image\_double - image\_mod\_3x3\_smooth;

k = 10;

image\_mask\_boosted = k .\* image\_mask;

image\_mod = image\_double + image\_mask\_boosted;

figure();

subplot(2,2,1)

imshow(image);

title("Before Transform");

subplot(2,2,2)

imshow(image\_mod\_3x3\_smooth);

title("After Smoothing");

subplot(2,2,3)

imshow(image\_mask\_boosted);

title(["Image Mask Boosted by k = " num2str(k)]);

subplot(2,2,4)

imshow(image\_mod);

title("Final Image");

Explanation: As seen in the final image, the craters and edges of the moon are greatly accentuated due to this effective high pass filter with the high pass part of the filter having a multiplier of x10.

3. Edge Detection

# Apply Sobel operators for edge detection and threshold to present the image in black and white for Fig4-3.tif

clear

fprintf("Question 3:")

file = 'Fig4-3.tif';

image = imread(file);

image\_double = im2double(image);

image\_mod\_3x3\_sobel\_x = zeros(size(image\_double));

image\_mod\_3x3\_sobel\_y = zeros(size(image\_double));

mask\_3x3\_sobel\_y = [[-1 -2 -1]

[ 0 0 0]

[ 1 2 1]];

for i = 1+1 : size(image\_double,1)-1

for j = 1+1 : size(image\_double,2)-1

frame\_3x3 = [[image\_double(i-1,j-1) image\_double(i-1,j+0) image\_double(i-1,j+1)]

[image\_double(i+0,j-1) image\_double(i+0,j+0) image\_double(i+0,j+1)]

[image\_double(i+1,j-1) image\_double(i+1,j+0) image\_double(i+1,j+1)]];

masked\_3x3 = frame\_3x3 .\* mask\_3x3\_sobel\_y;

image\_mod\_3x3\_sobel\_x(i,j) = sum(masked\_3x3,"all");

end

end

mask\_3x3\_sobel\_x = [[-1 0 1]

[-2 0 2]

[-1 0 1]];

for i = 1+1 : size(image\_double,1)-1

for j = 1+1 : size(image\_double,2)-1

frame\_3x3 = [[image\_double(i-1,j-1) image\_double(i-1,j+0) image\_double(i-1,j+1)]

[image\_double(i+0,j-1) image\_double(i+0,j+0) image\_double(i+0,j+1)]

[image\_double(i+1,j-1) image\_double(i+1,j+0) image\_double(i+1,j+1)]];

masked\_3x3 = frame\_3x3 .\* mask\_3x3\_sobel\_x;

image\_mod\_3x3\_sobel\_y(i,j) = sum(masked\_3x3,"all");

end

end

image\_mod\_3x3\_sobel = sqrt(image\_mod\_3x3\_sobel\_x.^2 + image\_mod\_3x3\_sobel\_y.^2);

image\_mod\_3x3\_sobel\_thresh = zeros(size(image\_double));

upper\_range = 255/255;

lower\_range = 64/255;

transform = 1:256;

for x = 1:256

if((x >= lower\_range\*255) && (x <= upper\_range\*255))

transform(x) = 255;

else

transform(x) = 0;

end

end

for i = 1:size(image\_double,1)

for j = 1:size(image\_double,2)

r = image\_mod\_3x3\_sobel(i,j);

if((r >= lower\_range) && (r <= upper\_range))

image\_mod\_3x3\_sobel\_thresh(i,j) = 1;

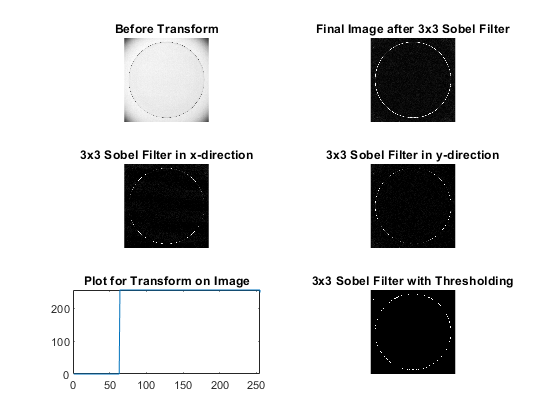
else

image\_mod\_3x3\_sobel\_thresh(i,j) = 0;

end

end

end



figure();

subplot(3,2,1)

imshow(image);

title("Before Transform");

subplot(3,2,2)

imshow(image\_mod\_3x3\_sobel);

title("Final Image after 3x3 Sobel Filter");

subplot(3,2,3)

imshow(image\_mod\_3x3\_sobel\_x);

title("3x3 Sobel Filter in x-direction");

subplot(3,2,4)

imshow(image\_mod\_3x3\_sobel\_y);

title("3x3 Sobel Filter in y-direction");

subplot(3,2,5)

plot(transform);

xlim([0,255]);

ylim([0,255]);

title("Plot for Transform on Image");

subplot(3,2,6)

imshow(image\_mod\_3x3\_sobel\_thresh);

title("3x3 Sobel Filter with Thresholding");