**Lab 1: Fundamentals of Image Processing**

****

Jonathan Tjong 20723414

William Li 20720929

**SYDE 575 - University of Waterloo**

Submitted To:Prof David Clausi

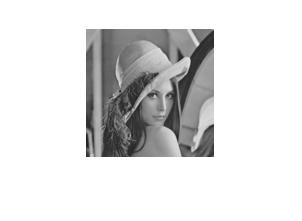
Due Date:Sept 21, 2021

# Introduction

In our introductory lab for Syde 575, we applied some of the concepts that we learned about in the week 2 content. Throughout the lab we were provided hands-on experience on digital zooming, image quality assessment, and point operations for image enhancement. As well, we practiced using MATLAB for image processing purposes. Throughout the lab, the following stock images were used: lena.tif, cameraman.tif, and tire.tif.

# 3. Digital Zooming

## Graphs and Images



| **Downsampled (Bilinear Interpolation)** |  |
| --- | --- |

**Upscaled Cameraman**

| Nearest NeighborPSNR: 21.5412 | Bilinear  PSNR: 21.8190 | Bicubic    PSNR: 22.2680 |
| --- | --- | --- |

**Upscaled Lena**

| Nearest Neighbor    PSNR: 26.6709 | Bilinear    PSNR: 27.2977 | Bicubic    PSNR: 28.0850 |
| --- | --- | --- |

### Q1 What can you observe about the up-sampled images produced by each of the methods?

All of the up-sampled images have some degree of information loss compared to the original. This is because the upsampling techniques can only interpolate and infer where information was lost, and cannot perfectly recreate the original image.

### Q2 How do the different methods compare to each other in terms of PSNR as well as visual quality? Why?

The images up-sampled using nearest neighbors have the lowest PSNR and as expected, have the lowest image quality. The nearest neighbor samples all have an aliased effect which can be seen by the “stair-casing” of pixels on rounded features in the picture.

Digital zooming by bilinear interpolation has the second highest PSNR and the second highest image quality. Bilinear interpolation has an anti-aliasing effect which is produced by interpolating the image values from the nearest 4 neighboring values.

Bicubic interpolation produced the highest PSNR and also produced the clearest image quality. Bicubic interpolation uses the same basic principle as bilinear interpolation, but interpolates from the nearest 16 neighboring values, which produces a more accurate upsampled image.

### Q3 What parts of the image seems to work well using these digital zooming methods? What parts of the image doesn’t? Why?

Parts of the image where there is little detail and flat surfaces work well for digital zooming. They work well as the information lost can be easily inferred by the surrounding areas.

Highly detailed portions of the image such as Lena’s hair don’t work as well for digital zooming. These areas are very complex and randomly chaotic so the digital zooming techniques can only infer what data was lost and cannot recreate what was exactly part of the original image.

### Q4 Compare the zooming results between Lena and Cameraman. Which image results in higher PSNR? Which image looks better when restored to the original resolution using digital zooming methods? Why?

All 3 Lena images showed higher PSNR values than any of the Cameraman images. The Lena images also looked better when restored compared to the Cameraman images. This could be due to the fact that there is a greater range of intensities in the Lena image, compared to the mostly dark Cameraman image. Since there is a greater range of intensities to infer from, the Lena image benefits more from the upscaling process than the Cameraman image. The Cameraman image is mostly dark to begin with, so no more clarity can be derived from the upscaling process.

### Q5 What does the PSNR tell you about each of the methods? Does it reflect what is observed visually?

PSNR shows the peak signal-to-noise ratio between a set of images. Generally, the higher the PSNR, the better the image recreation quality. By reducing the mean squared error between the recreation and the original, minimizing the difference between each pixel, the PSNR will be maximized. For each of the 3 upscaling techniques, the PSNR does reflect what is observed visually. The higher the PSNR, the more clear the upscaled image looked.

# 

# 4. Point Operations for Digital Enhancement

| **Original Tire image** |
| --- |
|  |

### Q6 Explain what the histogram of an image represents. Why is it useful?

The histogram of an image represents the frequency of each pixel occurring at any given intensity value (0-255). It is useful for determining the overall contrast levels of an image and for guiding the application of transformations to an image.

### Q7 Describe how the histogram looks like in the context of intensity distribution. What does the histogram say about the image?

Looking at the histogram, we can see that the tire image has a significant amount of low intensity pixels. Most of the intensity distribution is skewed to the left end; The low intensity values. This says that the tire image is overall very dark and black. This can be seen in the picture as the rubber tire portion is very dark and almost looks solid black. There is very little contrast here.

| **Negative transformation applied to tire image** |
| --- |
|  |

### Q8 Describe how the histogram looks like in the context of intensity distribution. How does it differ from the histogram of the original image? Why?

The negative image of the tire has lots of blown out whites which marks high on the intensity scale. As such, the histogram is heavily skewed to the higher intensity values. In comparison to the original tire image, the histogram is completely mirrored. This makes sense since the negative image should be the complete complement of the original image:

**Tire image after power transformations (**)

| **Gamma = 0.5** |  |
| --- | --- |
| **Gamma = 1.3** |  |

### Q9 Describe the appearance of the transformed images. Why do they appear this way?

After applying the power transformation with gamma = 0.5, the image became extremely dark and seemingly solid black. This is because output intensities are transformed into values that are significantly lower with a gamma value as small as 0.5.

After applying the power transformation with gamma = 1.3, the lighter areas of the image became very high intensity. As a result, lots of the picture area looks white. This is because the medium/high intensity values from the original image are transformed into a much higher value, in this case 255 (max intensity value, representing white).

### Q10 Describe how each histogram looks like in the context of intensity distribution. Why do they look like this? What does each histogram say about each transformed image?

For the histogram with gamma = 0.5, we see that the intensity distribution appears to be completely contained in the left side of the histogram. All the intensity values in the original image are transformed into much lower values due to the power law transformation. In fact, the maximum intensity value that could exist in the transformed image is .

For the histogram with gamma = 1.3, we see that the intensity distribution has an extreme skew to the right. In fact, the frequency of pixels with intensity 255 (white) is significantly greater than any other frequencies. This is because lots of the original intensity values are increased due to taking them to the exponent of 1.3, however with 8 bits we have a maximum of 255 intensity value. As such, every value in the range of in the original image is transformed to 255.

### Q11 Compared with the original image, which of the transforms should you use to enhance the image? Why?

Compared to the original tire image, a gamma correction of 1.3 should be applied. Since the tire image is mostly black with low intensity pixels, it will benefit from a gamma correction >1 as the gamma correction will compress lower intensities while stretching the high intensities. This can provide more contrast and help produce a more visually appealing image overall. Also, looking at the image after applying a transformation with a gamma value of 0.5, the image is dark and completely unrecognizable. So gamma correction of 1.3 is superior in comparison.

| **Tire image after histogram equalization** |
| --- |
|  |

### Q12 Describe the appearance of the equalized image

The equalized image looks more pale than the original, and much more detail can be seen on the tire itself. Nearly all the blacks and low intensity pixels have become more grey and the whites and high intensity pixels have been spread more evenly through the center of the tire.

### Q13 Describe how the histogram looks like in the context of intensity distribution. Why does it look like this? What does the histogram say about the equalized image?

### The histogram looks very evenly distributed. This means that the intensity levels were spread out and in the context of this image, we can observe that we have higher frequencies of medium intensities and high intensities, instead of having the majority of intensity values in the lower values. As a result our image has a much better contrast, providing higher quality detail. It is desirable to have a flat histogram since this represents the best possible contrast, and our histogram is closer to flat than it was before the equalization.

# Summary/Conclusion

In this lab, we explored three different methods of upsampling/digital zooming. We used nearest neighbours, bilinear interpolation and bicubic interpolation and compared the results of each method. To quantitatively compare each method, we calculated the PSNR (Peak Signal-to-Noise Ratio) of each set of images. The PSNR was calculated using the Mean Squared Error and a value of 255. By visual and PSNR comparison, it could be seen that bicubic interpolation produced the clearest upscaled image, bilinear interpolation produced the next clearest, and nearest neighbours the lowest quality upscaled image of the three.

Next we explored histograms and applying different transformations to an image. The first transformation applied was a negative image, which mirrored or complemented the original image’s intensity values. The resulting histogram was one that was completely flipped on the x-axis. Next, a series of power transformations were applied with different gamma values. The first gamma correction of 0.5 was applied. This increased the frequency of the lower intensities and made the image almost completely black. A gamma correction of 1.3 was also applied, which caused lots of intensity values to be increased to the maximum value of 255. As a result, the transformed image looks white in many areas. Finally, we performed a histogram equalization on tire image, which caused the intensities to be more evenly distributed, forming a flatter histogram. This resulted in the image having better contrast, revealing higher quality detail in comparison to the original image.

# 

# Appendix

Code:

**Lab1Part3.m**

% cameraman is already in grayscale

cam = imread('cameraman.tif');

% convert lena to grayscale

lena = rgb2gray(imread('lena.tiff'));

down\_cam = imresize(cam, 0.25,'bilinear');

down\_lena = imresize(lena, 0.25,'bilinear');

figure

imshow(cam);

figure

imshow(lena);

figure

imshow(down\_cam);

figure

imshow(down\_lena);

nearest\_cam = imresize(down\_cam, 4, 'nearest');

nearest\_lena = imresize(down\_lena, 4, 'nearest');

bilinear\_cam = imresize(down\_cam, 4, 'bilinear');

bilinear\_lena = imresize(down\_lena, 4, 'bilinear');

bicubic\_cam = imresize(down\_cam, 4, 'bicubic');

bicubic\_lena = imresize(down\_lena, 4, 'bicubic');

figure('name','nearest')

imshow(nearest\_cam)

figure('name','nearest')

imshow(nearest\_lena)

figure('name','bilinear')

imshow(bilinear\_cam)

figure('name','bilinear')

imshow(bilinear\_lena)

figure('name','bicubic')

imshow(bicubic\_cam)

figure('name','bicubic')

imshow(bicubic\_lena)

psnr\_nearest\_cam = psnr(cam, nearest\_cam);

psnr\_bilinear\_cam = psnr(cam, bilinear\_cam);

psnr\_bicubic\_cam = psnr(cam, bicubic\_cam);

psnr\_nearest\_lena = psnr(lena, nearest\_lena);

psnr\_bilinear\_lena = psnr(lena, bilinear\_lena);

psnr\_bicubic\_lena = psnr(lena, bicubic\_lena);

**mse.m**

function [mse] = mse(f,g)

%Mean Squared Error

% Assume f, g same size array

[row, col] = size(f);

mse = sum((double(f) - double(g)).^2, 'all')/(row \* col);

end

**psnr.m**

function [psnr] = psnr(f,g)

%Calculates PSNR from two images

% Detailed explanation goes here

mse\_out = mse(f,g);

psnr = 10\*log10(255^2/mse\_out);

end

**Lab1Part4.m**

% Close all open figures

close all;

tire = imread('tire.tif');

tire\_hist = imhist(tire);

figure

plot(tire\_hist)

title('Histogram')

ylabel('Frequency')

xlabel('Intensity')

figure

imshow(tire)

neg\_tire = 255 - tire;

neg\_hist = imhist(neg\_tire)

figure

imshow(neg\_tire)

figure

plot(neg\_hist)

title('Histogram of Negative Image')

ylabel('Frequency')

xlabel('Intensity')

% gamma = 1.3

temp = double(tire).^1.3;

power\_transform = uint8(temp);

power\_hist = imhist(power\_transform);

figure

plot(power\_hist)

title('Histogram of Image (Gamma = 1.3)')

ylabel('Frequency')

xlabel('Intensity')

figure

imshow(power\_transform)

equalized = histeq(tire)

equalized\_hist = imhist(equalized)

figure

imshow(equalized)

figure

plot(equalized\_hist)

title('Histogram of Equalized Image')

ylabel('Frequency')

xlabel('Intensity')