UNIVERSITY OF WATERLOO

SYDE 575

Group #22

Lab 1 Report

Prof David A. Clausi

Abdulhameed Naji - *20642623*

Shilpan Shah - *20603389*

Contents

List of Figures

1 Introduction

The purpose of this lab is to conduct a study of some fundamental image manipulation techniques such as digital zooming, spatial filtering, and point operations for image enhancement. Three images were used to carefully examine and study the effects of these techniques.

Digital zooming involved a quantitative approach to improving the perceived appearance of an image. A metric used for image quality assessment is Peak Signal to Noise Ratio (PSNR) which involves comparing individual pixels in two images. Performing the digital zooming experiment on two images showed that the Bicubic sampling method had the highest PSNR value, hence best image reconstruction. The method is followed by the Bilinear and Nearest methods respectively.

The discrete convolution experiment applied three filters or "masks" by convolving them with the source image. Two of the filters applied a directional blur by averaging nearby pixels whereas the other performed an edge detect operation by examining the difference between two pixels. It was concluded that convolution in 2D allows us to perform spatial filtering.

The point operation experiment examined an image histogram and performed an inverse and power transformation to manipulate the intensities and contrast in the image. Finally, the histogram was equalized, and the image contrast improved - hence concluding that a histogram of equal grey levels typically produces an optimal image contrast.

2 Digital Zooming

The experiment attempts to test the visual integrity of three sampling methods - Nearest Neighbor, Bilinear, and Bicubic. The source images shown in Figure 1 are down-scaled to $\frac{1}{4}$th the size as seen in Figure 2.

Figure 1: Comparison of source images

Figure 2: Comparison of down-sized images

The images are then up-sampled back to original size using the three methods. Comparison images of the results of each method are shown in Figures 3, 4, and 5.

Figure 3: Images upscaled by a factor of 4 using Nearest Neighbor Interpolation

Figure 4: Images upscaled by a factor of 4 using Bilinear Interpolation

Figure 5: Images upscaled by a factor of 4 using Bicubic Interpolation

2.1 Up-sampling Observation

Nearest neighbor displays more pixelated results, especially around high contrasting edges (cameraman outfit and aliasing on Lena's hat). This method is fast, but causes undesirable checkerboard effect. Bilinear results in a blurred appearance yet smoother result - some contrast information is lost as seen in the cameraman's equipment. Bicubic results are smooth and blurry, but captures contrasts and highlights better - Lena's hair now contains brighter highlights and some aspects of the cameraman image have sharper fall-off such as the spot captured in Figure 6 below.

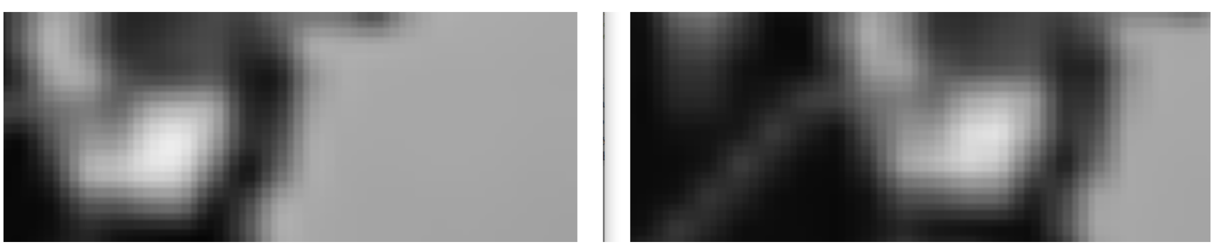

Figure 6: Zoomed in capture of the equipment in the cameraman image. The Bicubic method (Left) produces a sharper and brighter fall-off versus the smooth gradual results of the Bilinear method (Right)

2.2 PSNR Comparison

Figure 7 shows the resulting PSNR values by each method on both images. Recall a higher PSNR indicates a lower mean squared error - as such, higher PSNR correspond to a more accurate image reconstruction. In both images, the nearest neighbor method performed the worst - that is due to its naive sampling method. The values are binary and hence produce noticeable aliasing. The Bilinear method seeks to improve results by performing a simple linear interpolation to better represent the in-between colors. As a result, it performs a better reconstruction as indicated by the higher PSNR values. Finally, the Bicubic method reduces blurring by incorporating better contrast using its cubic curve fitting. As a result, Bicubic produces the best image reconstruction both visually and numerically.

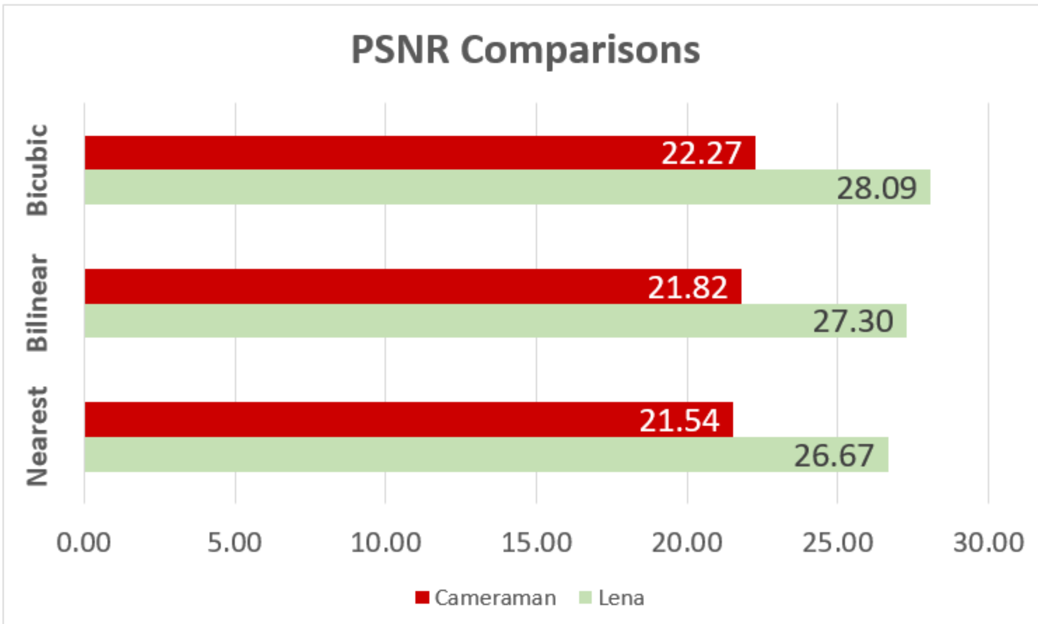

Figure 7: PSNR values shown for each method and image

2.3 Aspects Influencing Zooming

The bilinear method works best in areas of large detail. Finer, contrasting details are lost due to interpolation. This is evident in the cameraman equipment as the interpolation over a small area fails to capture large changes in intensity. On the other hand, the method produces more desirable results around edges as evidently seen in the camerman outfit.The bicubic method works marginally better on finer details as the interpolation naturally produces an ease in-out effect. As a result, fine details such as Lena's hair or the cameraman's equipment appear more "crisp" with more contrasting intensities. Parts of images that exhibit large features work better with the nearest neighbor method than finer details; the camerman image consisted of smaller details such as sloped lines that were aliased beyond recognition. Conversely, the Lena image suffers less in visual perception.

2.4 Cameraman versus Lena

The Lena image exhibits higher PSNR and is of better perceptual quality. The cameraman PSNR values were noticeably lower in all cases - the image suffers from greater artifacts such as aliasing and blurring due to its smaller starting size. Another contributing factor to the cameraman's lack of visual fidelity is its intensity composition. The range is biased to the darker intensities as seen in the histogram. The lack of contrast in the image makes it difficult for the human eye to identify features. Conversely, the features in the Lena image are finer and contain a broader range of intensities - hence better contrasting features. The lack of edge details or features is exacerbated when performing a re-sampling operation that would blur intensities. The range of colors is captured in the histograms of both images as shown in Figures 8 and 9.

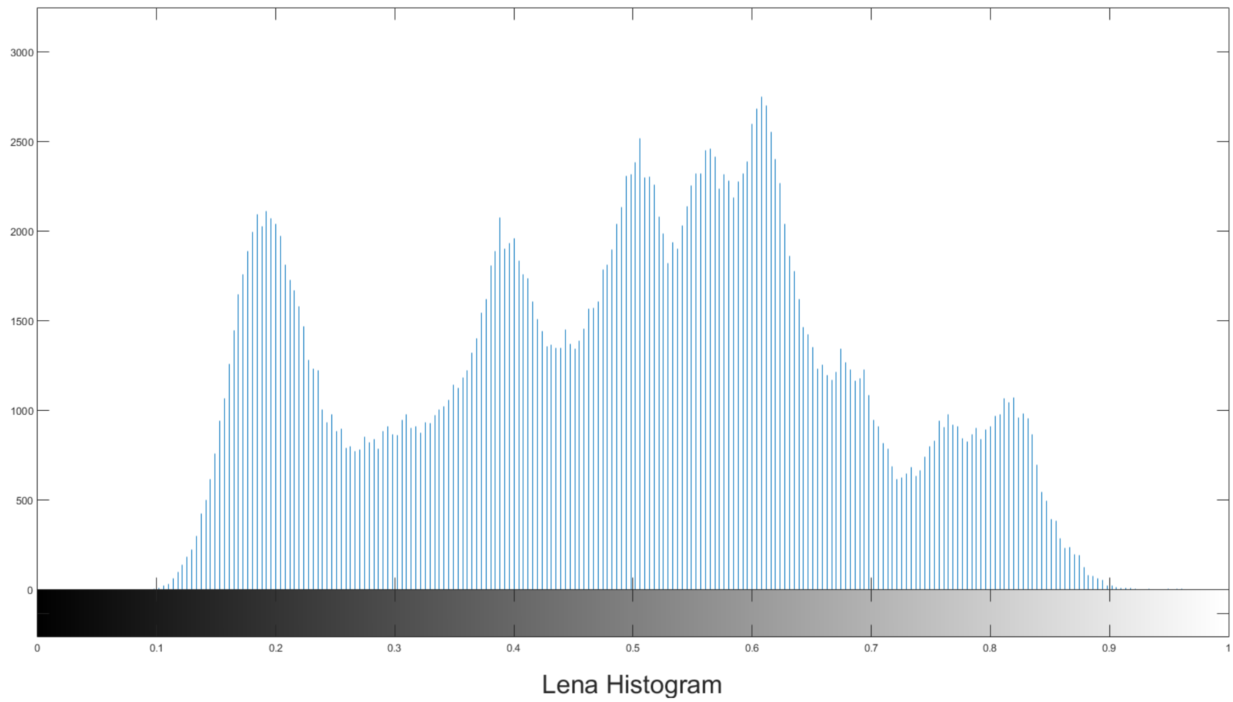

Figure 8: Lena image histogram

2.5 Conclusion of PSNR Values

The PSNR seems to indicate that the nearest neighbor performs the poorest image reconstruction - especially on low detail images such as the cameraman. The bilinear method is an improvement and works better for images that contain subtle intensity gradients. The method is not as effective on images such as the cameraman since we see a comparatively small improvement in PSNR as opposed to the Lena image. Bicubic performs best and combines the benefits of the former methods - it reconstructs better edge detail since interpolation occurs on a curve while also avoiding aliasing due to binary intensity sampling.

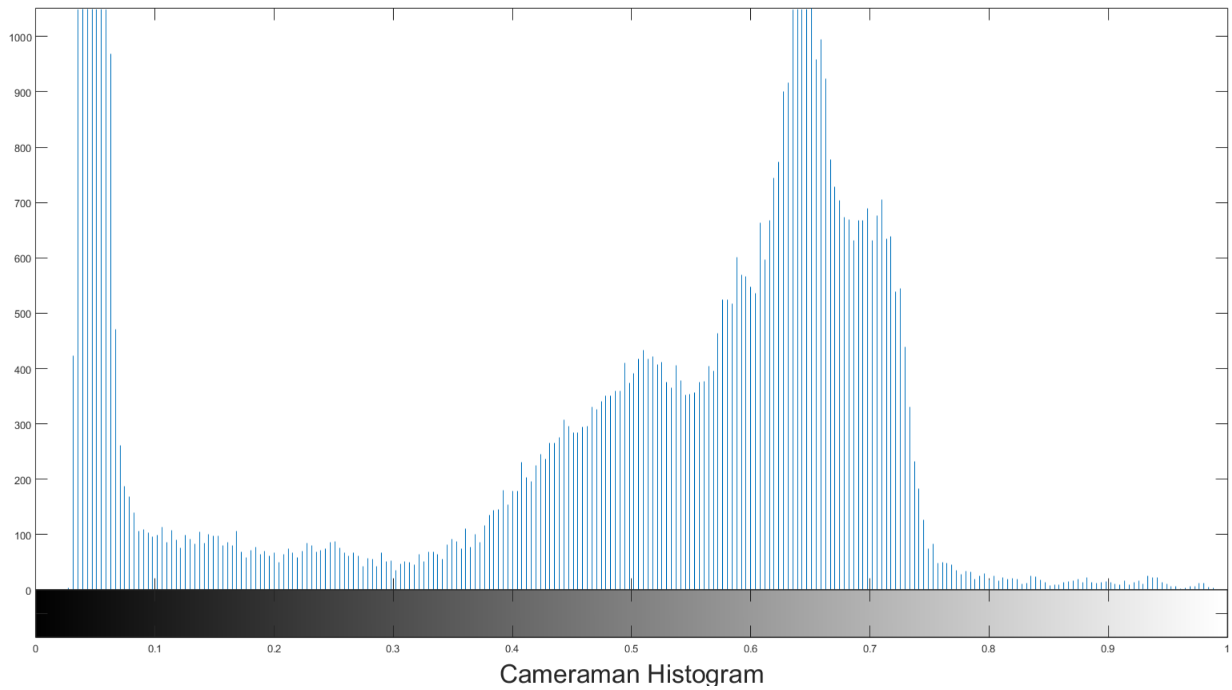

Figure 9: Cameraman image histogram

Generally speaking, nearest neighbor is not recommended for continuous images - instead, it is best used for categorical data such as grid/zone information where values need to be discrete and where new intensity generation must be avoided. Continuous data such as real life images are best handled by Bicubic and Bilinear interpolation. Bilinear can only produce values in between the two sampled intensities, whereas Bicubic may accidentally produce a value that is outside of range.

3 Discrete Convolution

This lab component attempts to take a source image and perform a convolution with predesignated vectors. The vectors are impulse functions of varying lengths and magnitudes. The image being convoled is the same Lena image shown in Figure 10.

3.1 Convolution with $h1$

The image was convolved with a filter that adjusted the intensity of the pixel by weighing contributions from 6 surrounding pixels in the horizontal direction. As such, h1 resembles a “mean filter” or ”moving average”; this explains the horizontal smearing artifacts since the pixel intensities will be blurred along the horizontal. Figure 11 shows the artifact due to the convolution.

Figure 10: Image subject to convolution

3.2 Convolution with $h2$

Since the filter is transposed, it corresponds to a change of direction. The mean filter now work on the vertical pixel direction - as such, the image appear smeared vertically. Figure 11 shows the artifact due to the convolution.

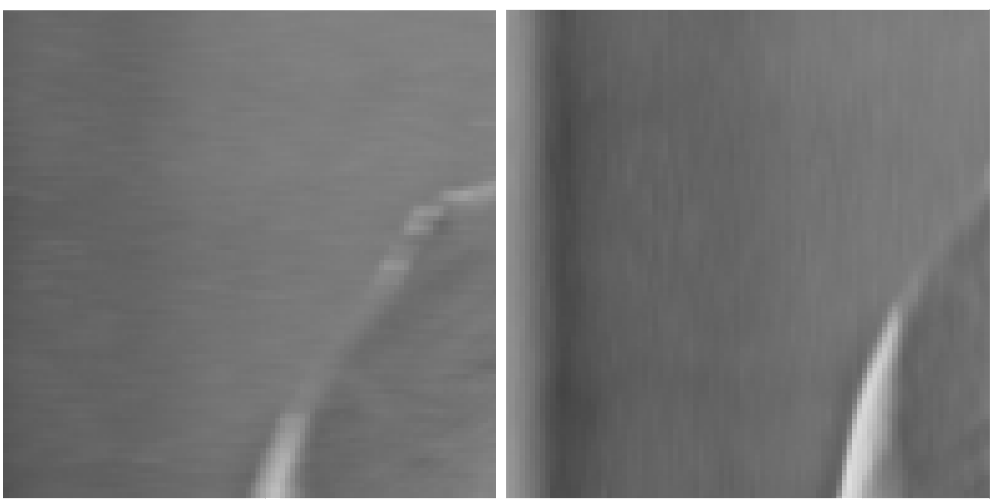

Figure 11: (Left) $h1$ filter versus (Right) $h2$ filter application

3.3 Convolution with $h3$

This filter looks at two neighboring pixels and calculates their difference. A large difference yields a higher intensity. As a result, the filter acts as an edge detector that only returns a bright intensity when it detects a large difference between neighboring pixels. Figure 12 shows the result of the convolution.

Figure 12: Convolution with $h3$ outputs the edges of the image shown in white

3.4 Interpretation of Convolution

2D Convolution allows us to perform spatial filtering - the operation sweeps the entire image; querying nearby pixels represented by the dimension of the filter vector/matrix. Neighboring pixels are assigned a weight that affects the final value of the current pixel in order to produce a desired result.

4 Histogram Operations

This lab experiment performs a set of histogram operations on the tire image.

4.1 Histogram Interpretation

4.1.1 Definition

A histogram is a probabilistic distribution of an image's grey levels. It describes the relative presence of a level or intensity in an image. It is useful because it allows us to directly characterize the image based on the intensities it contains.

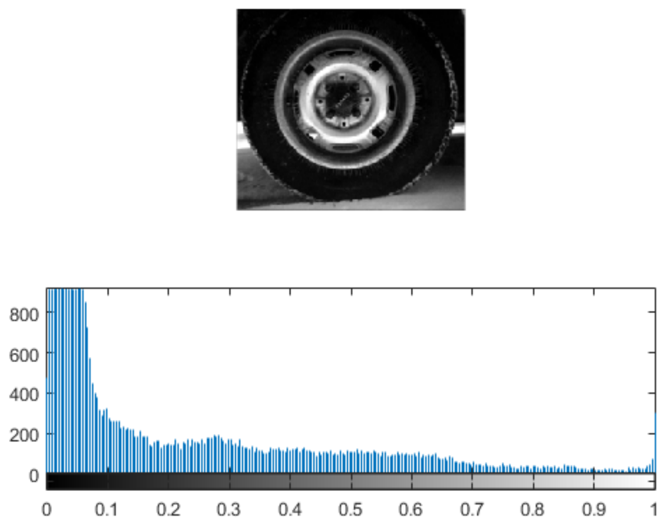

Figure 13: Source tire image histogram

4.1.2 Relation to Intensity

Each bar in a histogram represents the number of pixels (or weight if normalized) that is contributing to the bar's intensity range. Since the histogram describes the relative presence of intensity ranges, it can be used to infer contrast information, color correct images, or equalize intensities to achieve an optimal image quality. It tells us the quantity of "darkness" or "brightness" in an image. The dark intensities in the tire source image correspond to a histogram that is heavily biased to the darker ranges as shown in Figure 13.

4.2 Inverse Transform

Figure 14 shows the image after applying the inverse transform. The intensities are simply inverted (whites turn black and vice versa). As a result, the histogram is also mirrored to reflect the reversal of the image intensities.

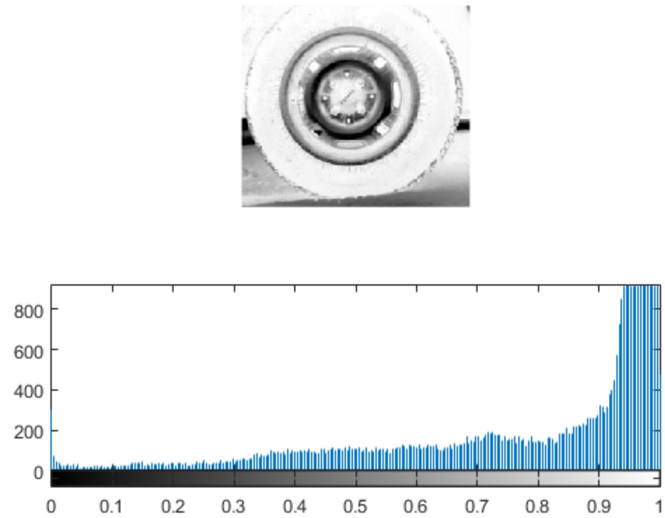

Figure 14: Inverted tire image histogram

4.3 Gamma Transform

Two gamma transformations are applied to the tire image and the respective results are shown in Figure 15.

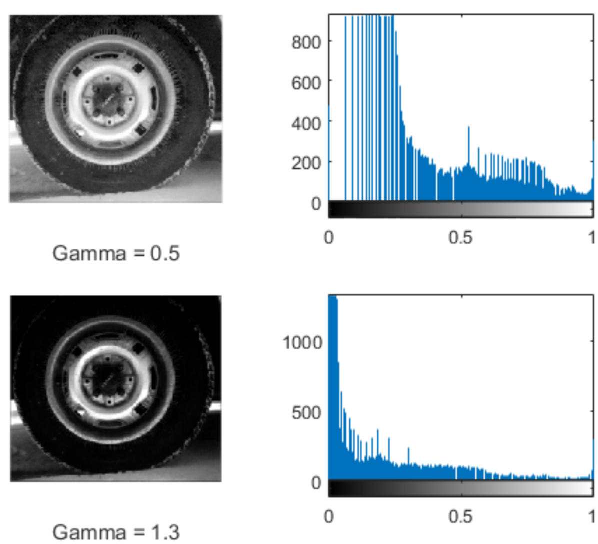

Figure 15: Gamma transformed tire images

4.3.1 Result Appearance

A gamma of 0.5 represents a square root function - such a function has an initially large slope before reaching an asymptotically low slope. The image consequently has its lower color ranges stretched. This means intensities in the lower range are being mapped to stretched, larger values. Hence, the image appears brighter with better contrast due to the broadened histogram. Conversely, a gamma of 1.3 is closer to linear than quadratic. The nature of this slope slightly compresses lower color ranges while stretching the higher intensity pixels. This means lower intensities have their range compressed, further increasing the darker contents of the image. The image consequently appears darker, with its histogram further concentrated on the lower intensity range.

4.3.2 Post-Transformation Histogram

The histogram for a gamma of 0.5 is broader; there is a broader presence of intensities. As such, the image appears with more contrast and brightness. Conversely, a histogram for a gamma of 1.3 is concentrated on low intensity range - there are more blacks than whites; the image appears darker and with less contrast.

The broader histogram is caused by low intensity range stretching due to the square root transformation. The low intensity concentrated histogram is the result of low intensity range compression that occurs due to the low slope value (< 1) present in the gamma function of value 1.3. Modification to the brighter color ranges is negligible because the function behaves close to linear at that range.

4.3.3 Comparison with Source

The transform with a gamma of 0.5 suits the image better - it stretches the darker spectrum and forms a broader color distribution. This provides better contrast and visual legibility.

4.4 Histogram Equalization

4.4.1 Equalized Appearance

The equalized image shown in Figure 16 has a broader color distribution which improves the image's contrast and brightness. There are details present that could not be perceived in the source image.

4.4.2 Intensity Contents

The histogram is now equalized - which means the relative contribution of each gray level is equal. Since all intensities are "equally" pronounced, the image has a better representation of bright and dark colors to form better contrast and visual perception. Since equalization requires all gray levels to have the same contribution, the histogram appears mostly flat.

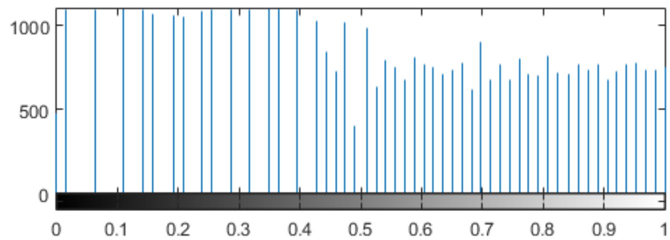

Figure 16: Tire image after applying histogram equalization

The non-ideality is a result of discretization, which forces the infinitesimal equalized bins to be discretely lumped into the next available bin - hence biasing the flatness of the histogram.

5 Conclusion

As seen in the lab report and results, the various techniques in image processing can be used to enhance image quality or manipulate the output of certain operations.

Digital zooming for the cameramen and Lena image involved using three methods for upscaling the downscaled image. The techniques were nearest neighbour, bilinear interpolation, and bicubic interpolation. It was found that bicubic method created greater fine details and reduced the checkerborad effect compared to the other techniques. Furthermore, bicubic interpolation method was found to have the highest PSNR values(22.27 for cameraman image and 28.09 for Lena image).

Convolution is used to perform spatial filtering by interpreting local information from neighboring pixels. The h1 and h2 filters convolved with the images result in an smeared or blurred image in the x and y direction respectively. The h3 filter calculates the difference between neighbouring pixel intensities; the larger difference then corresponds to a bright intensity that represents the edge in an image.

Histograms are used to describe the relative intensity level for an image. Using the histograms and concentration of pixel intensity, the darkness and brightness of an image can be found and compared. A dark image will have the intensities biased to the left side and bright image will have pixel intensities biased to the right side. This is shown in the image tires' histogram in Figure 13 and 14, where the inverse transformation is applied. Furthermore, when the gamma transformation is applied, the pixels are shifted towards higher or lower intensities depending on the gamma factor. When the gamma factor is 0.5, the lower grey levels are stretched and the higher grey levels are compressed. When gamma factor is 1.3, the the lower grey levels are slightly compressed whereas the higher grey levels are negligibly stretched, further darkening the image. Histogram equalization is used to flatten the histogram, where the relative contribution of each grey level is equal. After applying the histogram equalization, the tire features were more pronounced and the overall contrast improved.

6 Appendix

6.1 DigitalZooming.m

```
clear
clc

cameraman = double(imread('cameraman.tiff'));
lena     = (imread('lena.tiff'));
tire     = double(imread('tire.tif'));

gcameraman = cameraman;
glena     = double(rgb2gray(lena));
gtire     = tire;

%% Digital Zooming
MakeComparisonPlot(gcameraman, glena);
% Down Sizing
ds_cameraman = imresize(gcameraman, .25, 'bilinear');
ds_lena = imresize(glena, .25, 'bilinear');

dsPlot = MakeComparisonPlot(ds_cameraman, ds_lena);
dsPlot.Name = 'Down Sized Images';

% Nearest Neighbor
nn_cameraman = imresize(ds_cameraman, 4, 'nearest');
nn_lena = imresize(ds_lena, 4, 'nearest');

nnPlot = MakeComparisonPlot(nn_cameraman, nn_lena);
nnPlot.Name = 'Nearest Neighbor Result';

% Bilinear
bli_cameraman = imresize(ds_cameraman, 4, 'bilinear');
```

```
bli_lena = imresize(ds_lena, 4, 'bilinear');

bliPlot = MakeComparisonPlot(bli_cameraman, bli_lena);
bliPlot.Name = 'Bilinear Result';

% Bicubic
bc_cameraman = imresize(ds_cameraman, 4, 'bicubic');
bc_lena = imresize(ds_lena, 4, 'bicubic');

bcPlot = MakeComparisonPlot(bc_cameraman, bc_lena);
bcPlot.Name = 'Bicubic Result';

% PSNRs
nn_cameraman_PSNR = imPSNR(gcameraman, nn_cameraman);
bli_cameraman_PSNR = imPSNR(gcameraman, bli_cameraman);
bc_cameraman_PSNR = imPSNR(gcameraman, bc_cameraman);

nn_lena_PSNR = imPSNR(glena, nn_lena);
bli_lena_PSNR = imPSNR(glena, bli_lena);
bc_lena_PSNR = imPSNR(glena, bc_lena);
```

6.2 DiscreteConvolution.m

```
clear
clc

lena     = double(imread('lena.tiff'))./255;
glena    = rgb2gray(lena);

h1 = (1/6) * ones(1, 6);
h2 = h1';
h3 = [-1 1];

h1_lena = conv2(glena, h1);
h2_lena = conv2(glena, h2);
h3_lena = conv2(glena, h3);

h1Plot = MakeComparisonPlot(glena, h1_lena); h1Plot.Name = 'Lena * h1';
h2Plot = MakeComparisonPlot(glena, h2_lena); h2Plot.Name = 'Lena * h2';
h3Plot = MakeComparisonPlot(glena, h3_lena); h3Plot.Name = 'Lena * h3';

figure
imshow(glena);
```

```
figure
imshow(h1_lena);

figure
imshow(h2_lena);

figure
imshow(h3_lena);
```

6.3 PointOperations.m

```
clear
clc

tire     = double(imread('tire.tif'))./255;

%-----------------------------------------------------------------------
% Histogram: Tells use the distribution of intensity.
% If you don't know what this means, re-imagine the
% sad looking histogram for our class exam grades.
figure
    subplot(2,1,1);
        imshow(tire);
    subplot(2,1,2);
        imhist(tire);

% Negative Transform
%-----------------------------------------------------------------------
% Should be obvious what negative transform does. Simply inverts
% the values of the image. What was 1 becomes 0, and vice versa.
% As a result, the histogram is reversed.
negativeImg = 1 - tire;
figure
    subplot(2,1,1);
        imshow(negativeImg);
    subplot(2,1,2);
        imhist(negativeImg);

% Power Transform
%-----------------------------------------------------------------------
% The image contrast changes. The power function has the effect
% of crushing intensity values. 0.5 crushes whites and amplifies blacks.
% 1.3 crushes blacks and amplifies 0.5
% Makes sense mathematically, a low number raised to a power < 1 will
```

```
% attenuate it's value and vice versa.

% Power Transform - Gamma = 0.5
pwrImgA = tire .^0.5;
% Power Transform - Gamme = 1.3
pwrImgB = tire.^1.3;

figure
    subplot(2,2,1);
        imshow(pwrImgA);
        xlabel('Gamma = 0.5')
    subplot(2,2,2);
        imhist(pwrImgA);
    subplot(2,2,3);
        imshow(pwrImgB);
        xlabel('Gamma = 1.3')
    subplot(2,2,4);
        imhist(pwrImgB);


% Histogram Equalization
%------------------------------------------------------------------------
% Hypothetically, this should level off the spectrum of intensities in the
% histogram such that it's a flat line. What does this mean in terms of
% image perception? This should improve the contrast of the image because
% each intensity will have an equal contribution. In other words, there
% will be more whites and blacks and greys...more chance for colors to contrast.

equalizedImg = histeq(tire);

figure
    subplot(2,1,1);
        imshow(equalizedImg);
    subplot(2,1,2);
        imhist(equalizedImg);


\subsection{MakeComparisonPlot.m}
function outFigHandle = MakeComparisonPlot(imgA, imgB)
    if nargin < 2
        error('2 arguments are needed for images');
    end

    if max(imgA) > 1
        imgA = imgA/255;
```

```
    end

    if max(imgB) > 1
        imgB = imgB/255;
    end

    outFigHandle = figure;

    subplot(1,2,1);
        imshow(imgA);
    subplot(1,2,2);
        imshow(imgB);
end
```

6.4 imPSNR.m

```
function outPSNR = imPSNR( imgA, imgB )
%imPSNR Calculates the PSNR between 2 similar images
%   Given 2 equally sized images, calculate the PSNR
%   using the Mean Square Error (MSE).
imgA = double(imgA);
imgB = double(imgB);

[m, n] = size(imgA);
%meanSqr = (imgA - imgB).^2;
S=sum(sum((imgA-imgB).^2));

% Refer to MSE formula in lab manual
MSE = (1/(m*n))*S;

% We typically work with 8-bit images. Their max intensity
% representation is 2^8 = 255
MAX = 255;
outPSNR = 10*log10(MAX^2/MSE);
```