Image Processing
lab 1

Jan Boonstra (s2774100) Rick de Jonge (s2775832)

December 2, 2018

Exercise 1 – Reducing the number of intensity levels

a. In order to reduce the number of intensity levels of an image we first scale the grey-values of the input image between 0 and the desired amount of levels N. These numbers are then rounded, this divides the original pixels into N groups. We then scale the values back between 0 and 255 to obtain our output image. This code can be seen in Figure 1.

b. We used the script as seen in Figure 2 to produce the images with various levels.

```
%Reduces the number of intensity levels in Image from 256 to N
function out = IPreduce(Image, N)
    out = floor(N/256 * double(Image));
    out = uint8(out / ((N-1)/256));
end
```

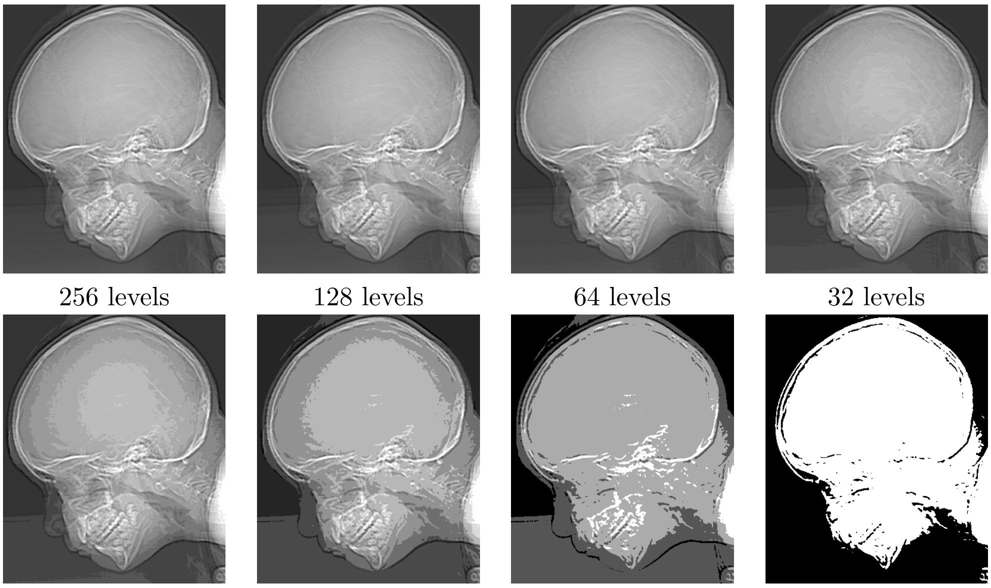
Figure 1: IPreduce.m

.

```matlab
close all;
clear all;

% Load skull image
skull = imread('../images/ctskull-256.tif');

% Reduce number of intensity levels
for Ncol = 2.^(1:8)
    skull_reduced = IPreduce(skull, Ncol);
    imwrite(skull_reduced, ['output/ctskull_reduced' num2str(Ncol) '.png'
        ]);
end

skull_reduced = IPreduce(skull, 4);

% Plot the input image
figure;
colormap(gray(256));
imshow(skull);
title('ctskull-256')

% Plot the reduced image
figure;
colormap(gray(256));
imshow(skull_reduced);
title('ctskull\_reduced')
```

Figure 2: Call the function IPreduce with the image named 'ctskull-265.tif'

.



256 levels 128 levels 64 levels 32 levels

16 levels 8 levels 4 levels 2 levels

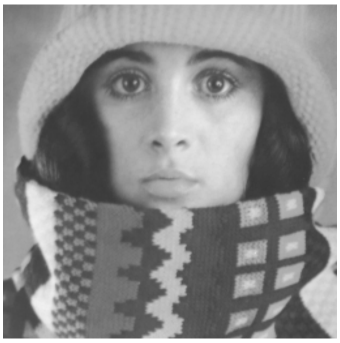
```
% Maximizes the contrast of Im (N-byte image) through linear stretching
function Im2 = IPcontraststretch(Im, N)
    M = max(Im(:));
    m = min(Im(:));
    c = (2.0^N - 1) / double(M-m);
    Im2 = c * (Im - m);
end
```

Figure 3: IPcontraststretch.m

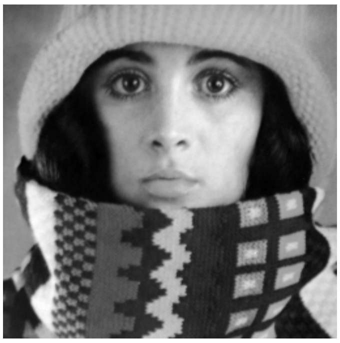
Exercise 2 – Enhancement

a. We used the formula $f_{out}(x,y) = \dfrac{2^8 - 1}{M - m}(f_{in}(x,y) - m)$ to scale the input image, ranging in grey-values from m to M, between 0 and 255. We used the code as seen in Figure 3.

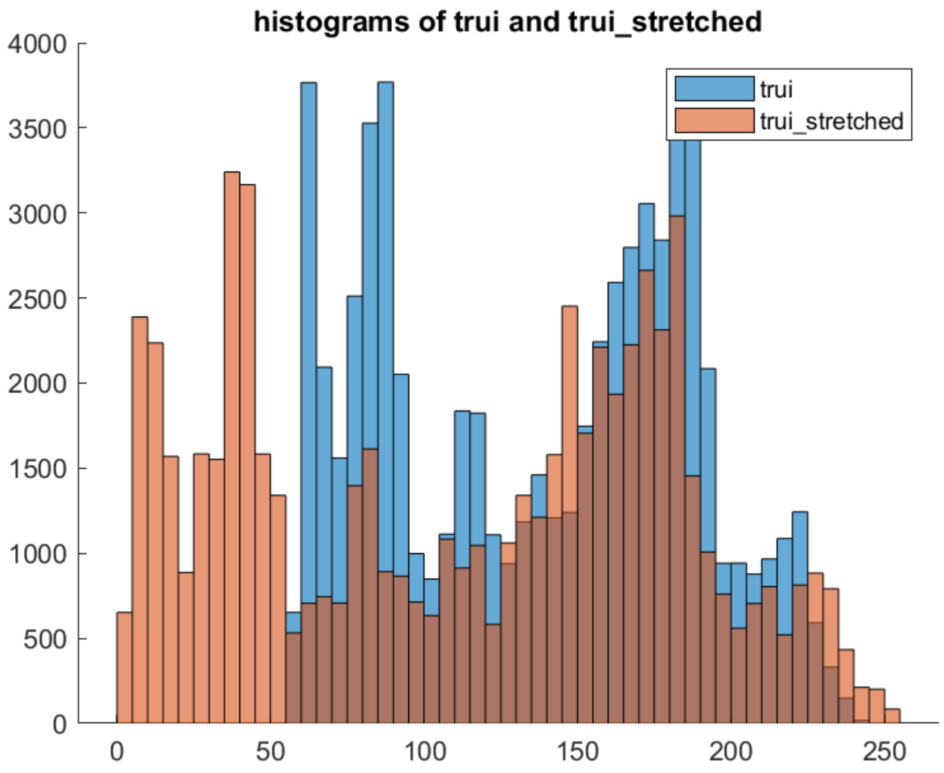
b. We used the script, seen in Figure 4, to produce the following results:



Original image



Scaled image



c. Since we are using the extreme grey-values now, the image will be darker in the areas that had grey-values between 0 and 127 and brighter where the values were between 128 and 255. We can also see this in the histogram, the areas on the far left and right are now occupied. These values came from the middle section, this looks like the middle part of the old histogram has widened to fit the whole range.

Contribution

The matlab code was written by Jan, while Rick supported (pair programming). The report was written by Rick, while Jan created the images and reviewed.

```
close all;
clear all;

% Load trui image
trui = imread('../images/trui.tif');

% Stretch contrast of image
trui_stretched = IPcontraststretch(trui, 8);
imwrite(trui, 'output/trui.png');
imwrite(trui_stretched, 'output/trui_stretched.png');

% Plot the input image
figure;
colormap(gray(256));
imshow(trui);
title('trui')

% Plot the stretched image
figure;
colormap(gray(256));
imshow(trui_stretched);
title('trui\_stretched')

% Plot histograms of the original and the stretched image
figure;hold on;
histogram(trui);
histogram(trui_stretched);
title('histograms of trui and trui\_stretched');
legend('trui', 'trui\_stretched');
saveas(gcf, 'output/histogram.png');
```

Figure 4: Call the function IPcontraststretch with the image named 'trui.tif'

.