**Introduction to Image Processing**

**Lab 3**

1. Test the following m code

%Quantize a signal to n bits. This code assumes the signal is between -1

%and +1.

n=8; %Number of bits;

m=120; %Number of samples;

x=sawtooth(2\*pi\*(0:(m-1))/m); %signal between -1 and 1.

%Trying "sin()" instead of "sawtooth"

%results in more interesting error(to the

%extent that error is interesting).

x(find(x>=1))=(1-eps); %Make signal from -1 to just less than 1.

xq=floor((x+1)\*2^(n-1)); %Signal is one of 2^n int values (0 to 2^n-1)

xq=xq/(2^(n-1)); %Signal is from 0 to 2 (quantized)

xq=xq-(2^(n)-1)/2^(n); %Shift signal down (rounding)

xe=x-xq; %Error

stem(x,'b');

hold on;

stem(xq,'r');

hold on;

stem(xe,'g');

legend('exact','quantized','error','Location','Southeast')

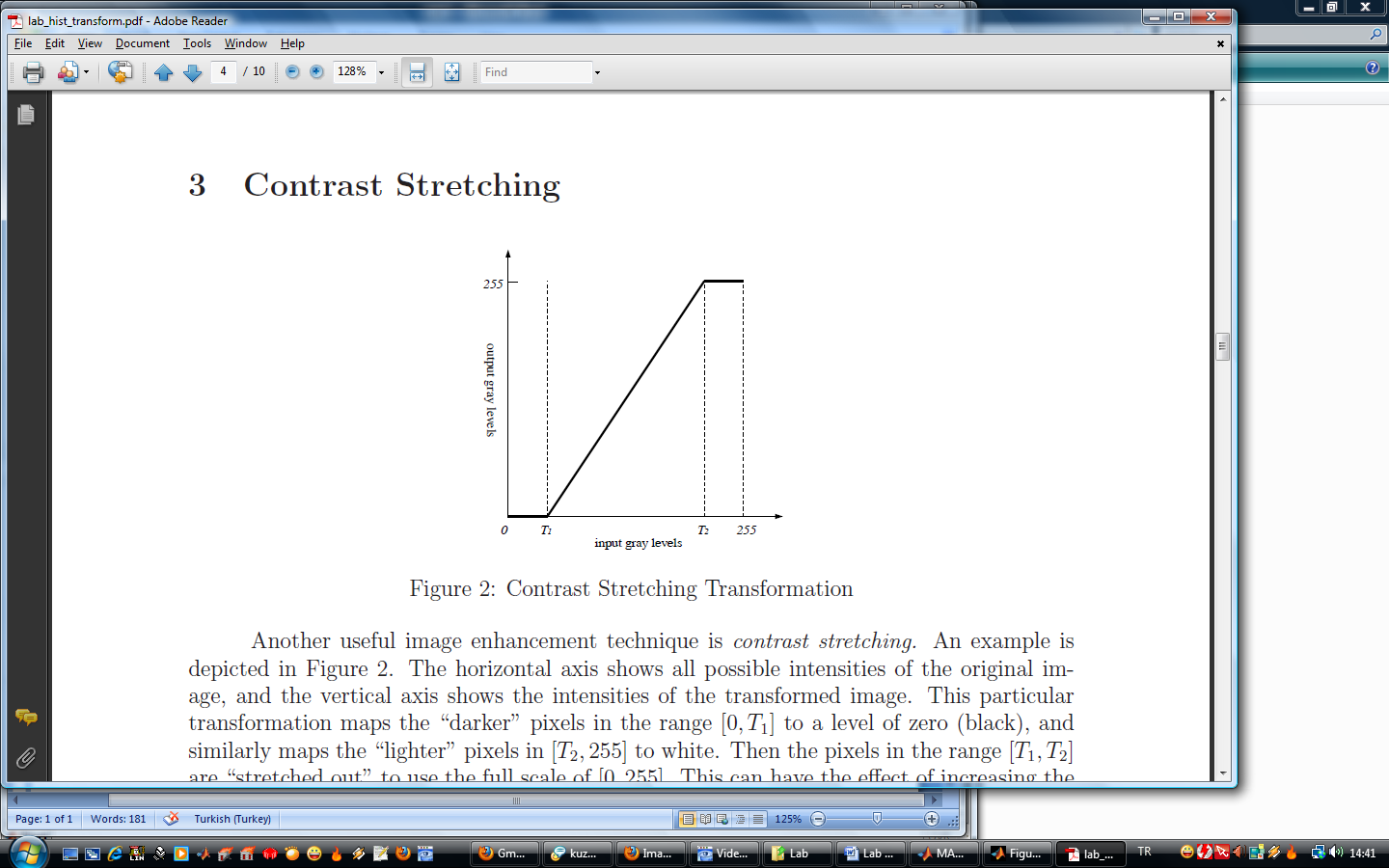
title(sprintf('Signal, Quantized signal and Error for %g bits, %g quantization levels',n,2^n));

hold off

* 1. Change the variable n and m to see the change of the output.
  2. Change the input x to different signal like sinusoidal or exponential and check the results.

1. Using the image *Rose1024.tif* obtain the subsampled versions of the image as given given in the presentation, i.e. obtain the images of 512x512, 256x256, 128x128, 64x64 and 32x32 rose images.

1. Perform thresholding on *moon.bmp* image to obtain the corresponding result in the presentation.
2. Use *Ctskull-256.tif* image to obtain different quantization levels of the image starting from 8 bits (original skull image), 7 bits, 6 bits, 5bits, 4 bits, 3 bits, 2 bits, and single bit images.
3. Use power law transformation with c=1 and γ = 0.6, 0.4, and 0.3 to obtain resulting images on *fractured\_spine.tif* image.
4. Write a Matlab function **output = stretch(input, T1, T2)** that will perform the pixel transformation shown in figure given below.



Use your **stretch** function to increase the contrast of the **kids.tif** image. Choose T1 and T2 so that the output image has a histogram which spans the full range of values from 0 to 255. Display the output image and its histogram.