BE 1. Image resizing and color enhancement

The objectives of this lab work are to understand

- how to resample an image using filtering and interpolation
- how to manipulate the contrast and colors of an image to make it more pleasant

Note: do it yourself! Do not use built-in functions of Matlab like imresize, rgb2hsv, etc.

Some useful commands:

```
I=double(imread('barbara.tif')); %to load an image
imshow(I/255);
colormap gray; axis image;


[H,W]=size(I);
J=zeros(size(I)); %J is a black image of same size as I
J=I>=100; %J is the binary image after thresholding I
```

Part 1 : Resampling

- 1) We want to reduce an image by a factor of two; that is, to transform an image of size N x N to a smaller image of size N/2 x N/2 pixels. This can be done by subsampling; that is, selecting every second line and every second column. Do this operation on the image **barbara**. Is the result satisfying? What kind of artifacts appears? To improve the result, the correct way of reducing an image is to perform lowpass filtering before subsampling. Based on the Shannon theorem, explain why lowpass filtering is required. Apply the lowpass filter **filter1=[-1/32,0,9/32,1/2,9/32,0,-1/32]** to the image, both horizontally and vertically, before subsampling. Comment on the result. Visualize the Fourier transform of the filter using **plot(abs(fftshift(fft([zeros(1,46),filter1,zeros(1,46)]))));** From this Fourier transform, explain why this filter is appropriate for lowpass filtering before subsampling by a factor of 2.

- 2) We want to enlarge an image by a factor of two; that is, to transform an image of size N x N to a larger image of size 2N x 2N pixels. For this, we upsample the image by inserting zeros between the pixels and then we apply the lowpass filter **2*filter1** to the image, both horizontally and vertically. Do this operation on the image **barbara** and comment on the result.

- 3) Run the file **image_rotation.m**. Open it and understand how it works. The method is based on interpolation and can be used for other problems. So, your goal is to create a similar procedure performing image enlargement of an image with an arbitrary factor f > 1, not necessarily an integer. For this, modify the file **image_enlargement.m.** Comment on the result on the image **cameraman.** In the case of an enlargement factor 2, compare the result on the image **cameraman** of this method and of the method of the previous question. Is the method appropriate for image reduction (if called with f<1)?

Part 2 : Modifying the color content of an image

- 1) Open the image **imcolor1.tif** and visualize it. The saturation is not satisfying: the colors are washed out, like if the image had been scanned from an old photograph. We want to increase the saturation, to make the colors more vivid and natural. First, transform the image **I** from the red, green, blue basis to the luminance, yellow-blue chrominance and red-green chrominance as follows:

```
IL=(I(:,:,1)+I(:,:,2)+I(:,:,3))/sqrt(3);
IC1=(I(:,:,1)+I(:,:,2)-2*I(:,:,3))/sqrt(6);
IC2=(I(:,:,1)-I(:,:,2))/sqrt(2);
```

Show how to reconstruct **I** from **IL, IC1, IC2**. Hint: the new basis is orthonormal, so that the matrix P=

1/sqrt(3)	1/sqrt(3)	1/sqrt(3)
1/sqrt(6)	1/sqrt(6)	-2/sqrt(6)
1/sqrt(2)	-1/sqrt(2)	0

is orthogonal: its inverse is its transpose.
Now, we will double the saturation of the image; for this, just multiply by two the images **IC1** and **IC2** before reconstructing the image in the R, G, B basis. Show the new image and comment on it.

- 2) Now we want to do the contrary. Open the image **imcolor2.tif** and visualize it. The saturation is too high and the image does not look natural. Divide the saturation by the appropriate factor and show the reconstructed image.

- 3) We want to modify the hue of an image to make some artistic effect. From the images **IC1** and **IC2**, we can define the saturation as **sqrt(IC1.^2+IC2.^2)** and the hue as **atan2(IC1,IC2)**. This may be viewed as a Cartesian to polar change of coordinates. Consequently, we can reconstruct **IC1** and **IC2** from the hue and saturation by **IC1=saturation.*sin(hue)** and **IC2=saturation.*cos(hue)**. Load the image **IM088.tif** and modify the image 1) by adding 0.6 to the hue and 2) by subtracting 0.6 to the hue. Show the results.