Lesson 8

Exam Part 2

Exercise - Part 1

1. Import "link.png" and "bird2.png" and store them in a dictionary;
2. Write a library with a sum_mask, noise and a medium filter function (the medium filter has not to count the edges of the image);
3. Perform the medium filter both on the starting images and the noised ones (use a loop in the dictionary);
4. Store these images into the same dictionary of the starting ones;
5. Save the 4 modified images (use a loop in the dictionary);
6. Send us by email:
   a. Both scripts (library and main)
   b. The 4 filtered images

Exercise - Part 2

1. Import the 6 images (original, medium, noise + medium x 2 images) of the previous part and store them into a dictionary (usa a loop);
2. Write a function that perform a <u>normalized</u> histogram (with relative frequencies, not the absolute ones), you must define the x-axis (use a binning of your choice) so that you will have histograms of the same length; Add this function to your library;
3. Apply it to all the 6 images and store them into a dictionary (use a loop);
4. Save the histograms DATA (use a loop or a compact syntax) on 2 files (1 file for the 3 histograms associated to each image); Each file must contain 4 columns: 1 - the x-axis, 2 - the y-axis of the starting image, 3 - the y-axis of the image with the medium filter, 3 - the y-axis of the image with noise + medium filter;
5. Plot the histograms (use a loop)- you should obtain 2 plot, in each one the 3 histograms associated to each image; Add legend, labels and so on;
6. Save the IMAGES;

Exercise - Part 2 optional

1. Write a function that perform the <u>derivative</u> of an image:
   a. Choose an axis (x or y or write it generically);
   b. Calculate the gradient (difference) between each pixel and the next/previous one (choose which one) creating the derivative matrix; Don't consider the edges of the matrix;
   c. Keep attention: your values will be in a range [-1, 1]. Check the minimum and the maximum number you have and renormalize your matrix.   New_val = (val - min) / (max - min)
2. Apply it to all the 6 images (store and save them ecc);