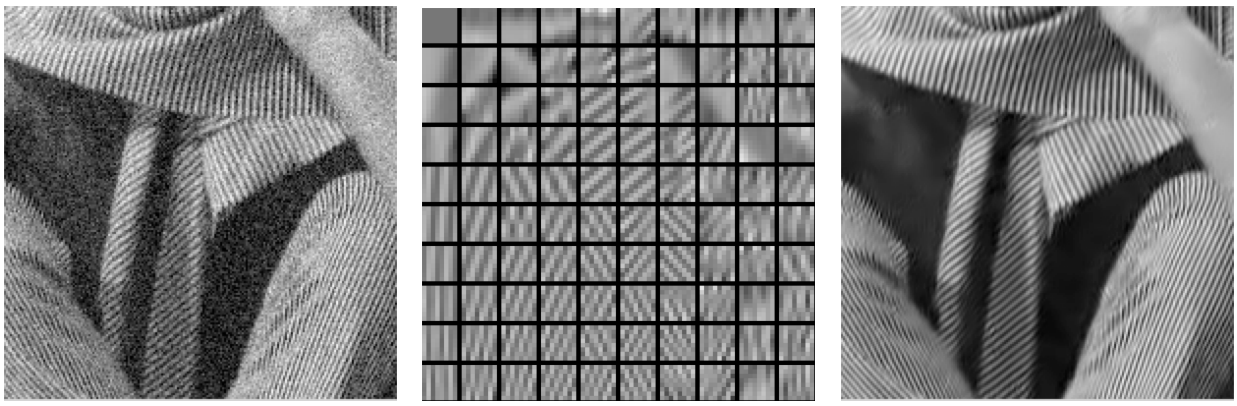
BIL 717 – Programming Assignment 3: Patched-based image denoising using learned dictionaries[1]

Due date: Monday, 2016-05-02, 11:59 PM.

Overview

The goal of this assignment is to get you familiarize with sparse modelling of images. In this assignment, you will use sparse representations to denoise a given image using a visual dictionary of patches.

Problem



(a) A noisy image (b) A dictionary of image patches (c) Denoised image

Figure 1: A sample denoising result.

You are provided some MATLAB files at the course webpage. You will use these files to denoise a given image as in Figure 1.

This code performs three different patch-based denoising algorithm which are respectively based on local-DCT w/o overlaps, local-DCT with overlaps, and K-SVD with overlaps. Moreover, it quantitatively evaluates the quality of the denoised images by means of the peak signal-to-noise ratio (PSNR) in DeciBels, which is defined as:

$$PSNR = 10log_{10}\left(\frac{255^2 NM}{\sum_{i=1}^{N}\sum_{j=1}^{M}(trueIm[i,j] - noisyIm[i,j])^2}\right) \quad (1)$$

Note that the higher PSNR is, the better the image quality.

All the provided models denoise a given noisy image by using a sparse representation of images. They consider visual dictionary which is fixed for the DCT-based models and which is trained from examples for the K-SVD model.

[1] Adapted from the tutorial given by Michael Elad, The Technion, Israel Institute of technology, Israel at Summer School on Sparsity in Image and Signal Analysis, Iceland, 2010.

Problem 1.1

In the first part of the assignment, your task is to analyze the provided denoising models. You should try the code with several images of your own and should play with the noise levels which is determined by `noiseSig`, the second argument of the function `Image_Denoising_Test.m` denoting the standard deviation of the noise.

Your experiments should be designed in a way that you can comment on the results obtained by the provided models and the effects of the two model parameters, `param.patchSize` and `param.nAtoms`, which respectively correspond to the size and the total number of visual words in the dictionary, on the denoising results. Please make sure to document and interpret your results in your reports.

Problem 1.2

In the second part of the assignment, your task is to transform the code provided to you so that the K-SVD model use a database of images to train a visual dictionary. That is, you should revise the code which learns a visual dictionary from the input image itself (`Train_Dictionary.m`) so that a visual dictionary is obtained from a set of provided images, which is in return used to denoise a given image.

You should analyse what the new model doing by examining the characteristics of the learned dictionaries, and their effects on the results both qualitatively and quantitatively. Please make sure to document and interpret your results in your reports.

Grading

The assignment will be graded out of 4: 0 *(no submission)*, 1 *(an attempt at a solution)*, 2 *(a partially correct solution)*, 3 *(a mostly correct solution)*, 4 *(a correct solution)*, 5 *(a particularly creative or insightful solution)*.

What to Hand In

You are required to submit all your code along with a report in HTML format. The codes you will submit should be well commented. Your report should be self-contained and should contain a brief overview of the problem, the details of your implemented solution and your comments about the experimental analysis on the aforementioned issues. Finally, prepare a ZIP file named `name-surname(s)-pa3.zip` containing

- `README.txt` - text file containing anything about the assignment that you want to tell but is not appropriate for the writeup webpage
- code/ - directory containing all your code for this assignment
- html/ - directory containing all your html report for this assignment, including images

The ZIP file will be submitted via email to `erkut@cs.hacettepe.edu.tr`.

Late policy

You may use up to five *extension* days (in total) over the course of the semester for the three programming assignments. Any additional unapproved late submission will be weighted by 0.5.

Academic Integrity

All work on assignments must be done individually unless stated otherwise. You are encouraged to discuss with your other classmates about the given assignments, but these discussions should be carried out in an abstract way. That is, discussions related to a particular solution to a specific problem (either in actual code or in the pseudocode) will not be tolerated. In short, turning in someone elses work, in whole or in part, as your own will be considered as a violation of academic integrity. Please note that the former condition also holds for the material found on the web as everything on the web has been written by someone else.