Final Project: Haar Wavelets –
NUMA01: Computational Programming with Python

Claus Führer

This assignment has 12 tasks.

The goal of the final project is to experience programming in a group. Proceed in the following way

- discuss the goal of the project in the group first until you fully understand the problem and get an idea for a practical approach
- divide the problem in subproblems
- discuss how the different parts should be tested
- prepare a max 15 min presentation which should present the background, the organization of your work, your solutions and maybe alternative attempts, which you decided to reject.

Bring your Laptop or an USB stick for the presentation.

Background

In this task we study a method for compressing a gray scale image. We want to compress it such that we can store it with as little data as possible as long as we can recognize the relation to the original. We don't care about quality otherwise.

The mathematical method we use is called Haar wavelets - a method based on simply taking arithmetic means and differences.

Part of the project is, that you read about the method yourself. The tasks guide you then through the steps of this project.

Before this, here some Python tools, which you might want to use:

- To have access to image tools in scipy:
  ```
  import scipy.misc as sm
  ```
- To read an image, convert it to grayscale and save it as an array:
  ```
  sm.imread('imagefile.jpg',True)
  ```
- To write an array `data` to a file as an jpg image:
  ```
  sm.imsave('newimagefile.jpg',data)
  ```

Tasks

The following tasks are guidelines to what you need to do. Feel free to do more, or modify them, as long as the spirit of the project remains the same.

Task 1

Read the text on Haar wavelets
http://www.whydomath.org/node/wavlets/hwt.html.
The sections on the analysis of HWT and Fourier series are not needed for this task.

Task 2

Download the picture `kvinna.jpg` from the webpage and save it as an array into an array.

Task 3

Determine the shape of the array. If the shape is not a tuple of two *even* numbers, delete a row, a column or both to make the shape a pair of even numbers.

Task 4

Compute to wavelet matrices one for the multiplication from the left and one for the multiplication from the right.

Task 5

Use these to make the wavelet transformation of the array. Save the resulting image in a file and look at it.

Task 6

The left upper subimage is the compressed image. The algorithm can be continued with the corresponding subarray in Python.

Task 7

Once you tried these steps out, meet with your group and design a program. You might want to create a function which does this Haar Transformation by

taking an array, cutting rows and columns if necessary and returning four smaller submatrices. These matrices correspond to the averaged picture and the three difference pictures. You might also want to write functions to generate the wavelet matrices.

Task 8

Write also a function `inverseHaarTransformation`, which takes the four submatrices, forms a big matrix and applies the inverse transformation.

Task 9

Finally write a program `Haarcompression`, which repeats this step on the upper lefthand submatrix.

Task 10

Test all these things also with a photo of your group.

Task 11

Discuss and implement a modification of your code which does the transformation directly by forming averages and differences without doing matrix multiplications. Compare computation time of both variants.

Task 12

Prepare a presentation with Powerpoint. This presentation should briefly introduce the algorithm, show some parts of the code you are proud of and finally demonstrate one example.

If you need consulting make an appointment!