Variational Methods for Computer Vision
Lecture: M. Möller
Exercises: J. Geiping
Winter Semester 17/18

Visual Scene Analysis
Institute for Computer Science
University of Siegen

Weekly Exercises 6

Room: H-C 6336
Friday, 08.12.2017, 14:15-15:45
Submission deadline: Tuesday, 05.12.2017, 14:15 in Room H-C 6336
Programmming: Email your solution to `jonas.geiping@uni-siegen.de`

Theory

Exercise 1 (4 Points: Poisson Editing Results in a Linear System)**.** Consider the Poisson editing approach discussed in lecture, i.e.,

$$\min_u \|\nabla u - \nabla g\|_2^2 \qquad \text{subject to } u(x) = f(x)\ \forall x \notin M,$$

for some inpainting domain M inside image f and a new image g that is going to be edited into f.
Show that the discrete version of the above problem reduces to solving a linear system.

Hint: Define a diagonal matrix D_M such that $(D_M u)_i = 0$ if $i \notin M$, and $(D_M u)_i = u_i$ otherwise. Can you incorporate the constraints by writing u as the sum of two parts which are multiplied with D_M and $I - D_M$ respectively?

Programming

Exercise 2 (4 Points)**.** Download the file 'exampleRawImageProcessing.zip' from the course website, in which you find parts of the data and code from `ftp://imageset@ftp.arri.de/`. Extract all files in one folder, start matlab and add the folder 'LUTs' to the path. Run 'demo.m' to get and impression of the processing chain.

Now you may either

- Implement an image denoising method right after the demosaicking,
- or implement bilinear demosaicking, e.g., by adapting the code from `https://de.mathworks.com/matlabcentral/fileexchange/5219-bilinear-interpolation?focused=6123822&tab=function` to the Bayer pattern layout 'grbg'.

You may take small crops of the original large input image. When implementing your method note that the default image format in the processing chain is uint16!

Exercise 3 (4 Points)**.** The goal of this exercise is to implement Poisson image editing based on your solution of exercise 2.

1. Find two fun images to fuse; if you don't know where to start, *www.pixabay.com* is a data base with many free-to-use images. Be aware that not all every image can be fused by Poisson editing, try some and see what works well.

2. Write a routine with which the user can interactively select a region in one image (e.g. using *roipoly*) and allow the user to select a location for this object in the second image.

3. Determine the inpainting domain M from the previous input, solve the linear system as computed in exercise 2, and display the final fused image.

Challenge: Be creative and create a fun fusion result! The results will be shown in the lecture. The funniest and most creative submission will receive a chocolatey award!