Multi-Exposure HDR Imaging and Tone Mapping

Due Date: 23:59 on Monday, June 10th, 2013

Figure 1: HDR imaging from multiple exposures of same scene

Background
Dynamic range of a scene is the ratio between largest and smallest values of a changeable brightness. Digital images are displayed on screen are 24-bits, that means 256 different intensity level can be shown. On the contrary dynamic range of human eye are far greater than 256 instensity levels so human eye can capture wider range of intensity than ones can be shown in digital images. This limitation of digital images can be solved by combining information from multiple exposures of the same scene as can be seen in Figure 1.

Overview
In this assignment you will implement Matlab procedures to create and display high dynamic range image by combining multiple exposures of same scene. For this purpose, you can use multiple exposure images given you or take own photos, but image you will take must be exactly the same scene with different exposures.

1Assignment and images adapted from Fredo Durand’s Computational Photography course
Details

Your programming assignment consists of the following two parts:

1. Building a HDR image from multiple exposures of the same scene

You will build a HDR image according to the equation below. Assume that given images are ordered with respect to exposure times, namely first image is the darkest image and the others are given in order of increasing exposure.

\[
out(x,y) = \frac{1}{\sum_{w_i(x,y)} \sum w_i(x,y)k_i I_i(x,y)}
\]

To implement this equation you must carry out steps below before.

- **Weight Computing**

  You will compute simple boolean weights \(w_i(x,y)\) for each image. You can write a Matlab function for this purpose. For an image \(i\), if the intensity value on \((x,y)\) is between 0.99 and 0.002, weight will be 1 else be 0.

- **\(k\) Multiplication Factor Computing Between Image Pairs**

  You can write a function for this purpose. Your function will take two images \((im1\ im2)\) and related weight maps \((w1\ w2)\) as inputs and return scalar \(k\) factor as output. You can compute \(k_i\) factor by using the following equation.

  \[
k_i = \frac{\text{median}(\frac{I_i(x,y)}{I_j(x,y)})}{k_j}
  \]

  for pixels \((x,y)\) st. \(w_i(x,y) > 0\) and \(w_j(x,y) > 0\)

  After you compute \(k\) factors for pairs of images, you will compute \(k_0\) and chain it to factors.

- **Special cases:** Some pixels might be underexposed or overexposed in all images. You should only threshold in one direction on darkest and brightest image.

2. Displaying HDR image: Tone Mapping

You will write a Matlab function that takes HDR image as input and produce RGB image tone mapped from HDR image. Your function prototype must be as follows:

\[
RGB = \text{toneMap}(HDR, \text{filename})
\]

You will do the following processes inside of this function.

(a) Computing the intensity \(I\): You can average RGB channels. \(I = 0.4R + 0.7G + 0.01B\)

(b) Computing the chrominance: \((R/I, G/I, B/I)\)

(c) Computing log intensity: \(L = \log_{10}(I)\).

(d) Filtering log intensity: \(S = \text{bilateral}(L, \sigma_1, \sigma_2)\) or \(S = \text{gaussian}(L, \sigma)\)

(e) Computing high frequency part: \(H = L - S\)

(f) Contrast reduction in log domain:

\[
\text{range} = \max(S) - \min(S)
\]

Scale factor \(k = \log_{10}(100)/\text{range}\) for 1:100 contrast

\(S' = k(S - \max(S))\)

(g) Reconstruction log intensity: \(L_{new} = 10^{(S' + H)}\)
(h) Combining color information: $RGB = L_{new} \ast (R/I, G/I, B/I)$

(i) Applying gamma compression: $RGB^{0.5}$

Some important remarks:
- You can use Fast Bilateral filter code here: http://groups.csail.mit.edu/graphics/bilagrid/
- You must compare and comment Bilateral and Gaussian filter effects
- You must show results for at least 3 different images in your report.
- You can test your results by comparing Matlab built-in function results.

Grading

The assignment will be graded out of 100 points:
- 0 (no submission), 20 (an attempt at a solution), 40 (a partially correct solution), 60 (a mostly correct solution), 80 (a correct solution), 100 (a particularly creative or insightful solution)

Note: Preparing good report is important as well as your solutions!

What to Hand In

You are required to submit all your report along with a short webpage in HTML. For that purpose, prepare a folder containing
- HTML/README.txt (text file containing details about your project)
- HTML/code/ (directory containing all your code)
- HTML/ (directory containing all your documents, including your images)
- HTML/data/ (including your data images)
- HTML/result/ (including your result images)
- HTML/index.html (html report)

Archive this folder as studentid_pset4.zip and send to karacan@cs.hacettepe.edu.tr.

Each student must individually do the coding and prepare detailed report which contains a brief overview of the problems, details of implementation and the results with comments. All results you obtain must be put to your report and observations must be specified clearly. If your implementations failed to give a satisfactory results, provide a brief explanation of the reason(s).

Policies

All work on assignments must be done individually unless stated otherwise. You are encouraged to discuss with your 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 else's work (from internet), 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.