Problem Set 3

Assigned: March 9, 2006
Due: March 23, 2006

Problem 1 (Optional) Multiple-Exposure HDR Images

Even though this problem is optional, we recommend you to go through the capturing process and create your own HDR image. This can be done in groups of up to three people. Please identify your collaborators.


Calibrate the camera response curve

You need many pictures for this job (e.g. 10 every 1/3 stop).

• Use manual mode (M).
• Set small aperture.
• Vary shutter speed in 1/3 stops, that is, one step at a time (e.g. from 1/10 to 1/13 to 1/16 to 20, etc.) Take maybe 10 images from underexposed to overexposed. Avoid moving the camera when you change the shutter speed. Of course, use the tripod.
• Save images on computer and go to HDRShop.
• Go to menu Create—>Calibrate camera curve.
• Select image sequence (you might need to select them in small groups).
• Indicate increment between images (click on 1/3 F-stop).
• Click Go.
• When the curve looks nice, press stop and save curve.
Use the bracketing mode on the camera to take a sequence of 3 pictures

Find a scene with high contrast. A good example is a scene with both indoor and outdoor parts. Check that the scene is too contrasted by taking a picture and checking on the back LCD that part of it is under- or over-exposed.

On a Nikon, Press BKT (top left) and rotate the back dial to set it on. Put the top-left dial on "C" (continuous) so that when you keep the shutter pressed, it takes the series of 3 pictures. On a Canon, you need to go somewhere in the menu. Use a remote shutter release to avoid camera shake if possible.

Assemble a high-dynamic-range image using HDRShop

- Create→Assemble HDR from image sequence.
- Load images, indicate increment (e.g. 3 stops).
- Change camera response curve (custom curve, Browse to select your new curve).
- Press generate Images.
- Play with the view menu, in particular with exposure.
- Save as .hdr

Problem 2  Tone Mapping Using the Bilateral Filter

We now want to reduce the dynamic range of our image to display it on a low-dynamic range device. For this, you will implement a simplified (read: slow) version of Durand and Dorsey’s 2002 algorithm. This algorithm only modifies the luminance of an image: it reduces the contrast of the large-scale variation of luminance but preserves local detail. For this, it decomposes the luminance image into a large-scale (a.k.a. base) layer and a detail layer using the bilateral filter. All computation on luminance is performed in the log domain.

The bilateral filter blurs an image except across strong edges. For each pixel, the output is a weighted average of the neighboring pixels where the
weight depends on both the spatial distance and the intensity difference:

\[
J_s = \frac{1}{k(s)} \sum_{p \in N_s} \exp\left\{-\frac{(p - s)^2}{2\sigma_d^2}\right\} \exp\left\{-\frac{(I_p - I_s)^2}{2\sigma_r^2}\right\} I_p
\]

where \(k(s)\) is a normalization term

\[
k(s) = \sum_{p \in N_s} \exp\left\{-\frac{(p - s)^2}{2\sigma_d^2}\right\} \exp\left\{-\frac{(I_p - I_s)^2}{2\sigma_r^2}\right\}.
\]

\(N_s\) is the neighborhood of \(s\). \(s\) and \(p\) are both coordinates of image lattice. So there are three parameters for bilateral filtering, the half size of the neighborhood \(w\) (the neighborhood is \((2w + 1) \times (2w + 1)\)), spatial standard deviation \(\sigma_d\) and range standard deviation \(\sigma_r\).

Here is the pseudo code of applying bilateral filtering for tone mapping.

```matlab
input intensity = 1/61*(R*20+G*40+B)
r=R/(input intensity), g=G/input intensity, B=B/input intensity
log(base)=Bilateral(log(input intensity))
log(detail)=log(input intensity)-log(base)
compressfactor=log(output range)/(max(log(base))-min(log(base)));
log_offset=-max(log(base))*compressfactor;
log(output intensity)=log(base)*compressfactor+log_offset+log(detail)
R output = r*exp(log(output intensity)), etc.
```

The main parameter of this code is output range, which depends the amount of remaining large-scale contrast that you want in the output. A value of 10 to 30 works well.

(a) Write a MATLAB function \(Im=imbltflt(im,wsize,sigma_d,sigma_r)\) to implement bilateral filtering. Argument \(im\) is the input image. Parameter \(wsize,\) \(sigma_d\) and \(sigma_r\) correspond to \(w, \sigma_d\) and \(\sigma_r\) in the above equations. Try to avoid writing four loops. Two loops (over image coordinate) are OK. Load image \(einstein.jpg\), add AWGN with \(\sigma = 0.05\) (as you did in ps2), and apply bilateral filter to denoise the noise contaminated image. Use parameter \(w = 5, \sigma_d = 2, \sigma_r = 0.12\). Display the noisy image and denoised image.

(b) Load HDR image \(vinesunset.hdr\) using the enclosed MATLAB code \(read_rle_rgbe.m\) (from http://www.cis.rit.edu/mcsl/icam/hdr/). Implement tone mapping using Gaussian filtering, \(\text{fspecial('gaussian',21,8)}\),
instead of bilateral filtering. Set output range to be 30. Display the LDR image. Do you see any artifacts?

(c) Now replace the Gaussian filter with the bilateral filter. Set the parameter $w = 10, \sigma_d = 8, \sigma_r = 0.2$. Display the LDR image. Do the halo artifacts disappear? Compare the result with (b).

(d) (Extra credit) Implement the uncertainty “fix”; implement fast bilateral filtering; implement the trilateral filter

---

**Problem 3  Poisson Image Editing**

Your goal is to create a photomontage by pasting an image region onto a new background using Poisson image editing. Please read Pérez et al’s *SIGGRAPH* 2003 paper “Poisson Image Editing”. Focus on Section 2 and 3.1. Implement the algorithm from Equation (7) to (11).

The main task of Poisson image editing is to solve the huge linear system $Ax = b$ in Equation (7). You can explicitly represent $A$ using sparse matrices, and solve it using MATLAB command \. This method should work for the provided examples, but cannot scale up for big masks. We do not recommend this approach, though it is fine for you to use it.

We recommend conjugate gradient (CG) approach. The basic idea of conjugate gradient method is that the solution to $Ax = b$ can be represented by a set of vectors, and these vectors are $A$ orthogonal between one and another. It is mandatory that matrix $A$ is positive definite. In energy minimization problems $A$ is always semi-positive definite. In our problem matrix $A$ is the second-order derivative matrix which is positive definite.

There are two reasons that (preconditioned) conjugate gradient method is favored in image processing/editing

- For image processing matrix $A$ is huge. The space complexity for LU decomposition is $O(n^3)$, too big for images. But CG only requires matrix multiplication in each iteration, which is essentially filtering. Space complexity is $O(n^2)$

- CG gives good results within a few iterations. Normally the result is good enough if the number of iteration is image width or height, or even less. So CG is efficient. It can be much faster, converging in less
than ten iterations, if a good preconditioner is used (not required for the homework).

The conjugate gradient method can be summarized in the following pseudo code

Given $A$, $b$ and an initial guess of $x$, solve $Ax=b$ iteratively

1. $r = b - Ax$  \(\%\) residual
2. for $k = 1 : n$
   \begin{itemize}
   \item $\rho_k = \|r\|^2$
   \item If $k = 1$ then $p = r$ else $p = r + \frac{\rho_k}{\rho_{k-1}}p$ \(\%\) direction
   \item $q = Ap$
   \item $\alpha = \frac{\rho_k}{(p^Tq)}$
   \item $x = x + \alpha p$ \(\%\) new point
   \item $r = r - \alpha q$ \(\%\) residual
   \end{itemize}
3. Output $x$

Your task is to replace $Ax$ by filtering and other MATLAB operations. For this example you will get good enough result with $n = 100$. To debug you can monitor the norm of $r$ which should be monotonically decreasing.

(a) Load image `bear.bmp` and a binary mask `mask.bmp`. Seamlessly paste it onto image `waterpool.bmp` so that the top-left coordinate of the mask is (20,150) at the background image. Try also (20,1) and see how the results are different.

(b) So far we have implemented three methods for seamless pasting, i.e. multi-scale blending, matting and Poisson image editing. Compare the three methods in terms of advantage, limitation, speed, application scenario and so on. You may list a table and describe in three paragraphs.

**Problem 4 \(6.882\) only Paper Review**

Go to page [http://groups.csail.mit.edu/graphics/classes/CompPhoto06/papers.htm](http://groups.csail.mit.edu/graphics/classes/CompPhoto06/papers.htm). It is also included in the zip file. Choose a paper from the literature among the ones proposed on the web page. Alternatively, a different paper can be used with permission of the instructors. Use the SIGGRAPH review form in the zip file and evaluate the article.