

Stephen Whittaker  
SP.747 – Creative Imaging  
Final Project – High Dynamic Range Imaging

Have you ever wanted to shoot an extremely high contrast image and not been able to expose all of the scene correctly? This happens a lot in photography. If you're inside and you want to take a picture near a sunny window, you have to choose which parts of the image you want to be exposed correctly. You can choose the inside and have the window be extremely overexposed, or you can choose the outside and have the inside extremely underexposed. With the advent of digital cameras, a relatively new photographic technique has emerged that attempts to solve this problem : High Dynamic Range Imaging.

High Dynamic Range Imaging, or HDRI for short, allows a series of photographs containing much more information than a single photograph to be merged into one. Each exposure can have a different region of the scene correctly exposed. Once the three have been merged, the dynamic range of the merged photo can be compressed. This will allow a High Dynamic Range image to be displayed on a medium that is incapable of rendering raw HDR images.

For my final project, I experimented with the new HDR capabilities in Photoshop CS2. To start, I took three separate photos of the same, very high contrast scene. The picture was taken from the roof of my house in Boston, right by the Citgo sign.



**Taken at the camera's estimation of correct exposure.**



**Taken at -2.0 EV from the first picture**



**Taken at +2.0EV from the first picture**

These three images contain a significantly higher amount of data than the first picture by itself. Now, we can merge the images using the “Merge to HDR” action in Photoshop CS2. This is under the File -> Automate menu. A 32-bit HDR image will result from this merge.

Once the images are merged, they look very similar to the original. This is because a computer monitor is incapable of displaying a raw HDR image. The dynamic range must be compressed so that is in the viewing range of the medium. This is done by converting the 32bit image to an 8bit image. You can do this in the Image->Mode menu and click on 8bits/ channel.

This is perhaps the hardest part of the entire process – you have to map the 32bit values into the 8 bit values we see on screen. It works very similar to the “curves” feature under the adjustment menu. One axis is the input image’s range of values and the other axis is the output image’s range of values. To make this whole process easier, there are plugins that use complex algorithms to do this process automatically. Photomatix’s ToneMapping plugin is a good example and is what I used here.



**The final, correctly exposed image**