Project 4: Image Based Lightening

Points Estimate:
I am expecting a total of 100 Points for project4.

1. Recovering HDR Radiance Maps (50 Points)
    - Data Collection (10 Points)
    - Naive LDR merging (10 Points)
    - LDR merging without under and over exposed regions (10 Points)
    - LDR merging and response function estimation (15 Points)
    - Irradiance Discussion

`

2. Panoramic transformations (10 Points)
3. Rendering synthetic objects into photographs (30 Points)
4. Quality of results and Project page

Bells and Whistle:
I am expecting a total of 20 Points from Bells and Whistles.

20 Points for Additional Image-Based Lighting Results
- Presented a new objects with same HDR light map
- Composed results with a new HDR light map

Summary:
For project 4, I worked with two scenes.
To begin with I use the scenes provided in the starter code as a reference and accomplished all the required tasks required to get 100 Points.
Second scene is create at my workplace in my home. I used 6 objects from the blender (Try finding the monkey) and experimented with different kinds of materials.

Samples folder has the results generated by executing the code with sample/starter code provided scene images.
“images/3/” has the results generate by executing the code with scene I created.

Recovering HDR Radiance Maps (50 Points)
Data Collection (10 Points)
Sample data from the starter program is used for reference.
Sample data exposures used are 1/24, 1/120, 1/553. These images are staged up into ldr_images with a corresponding exposures value stack. We use these values for future calculations.

Naive LDR merging (10 Points)
Naive LDR merging is calculated by dividing intensities of the pixels with its exposure values of every ldr_images and averaging the resultant image
Log irradiance is calculated by taking a logarithmic difference of intensities of the pixel and the exposures of that ldr_image and normalizing it.

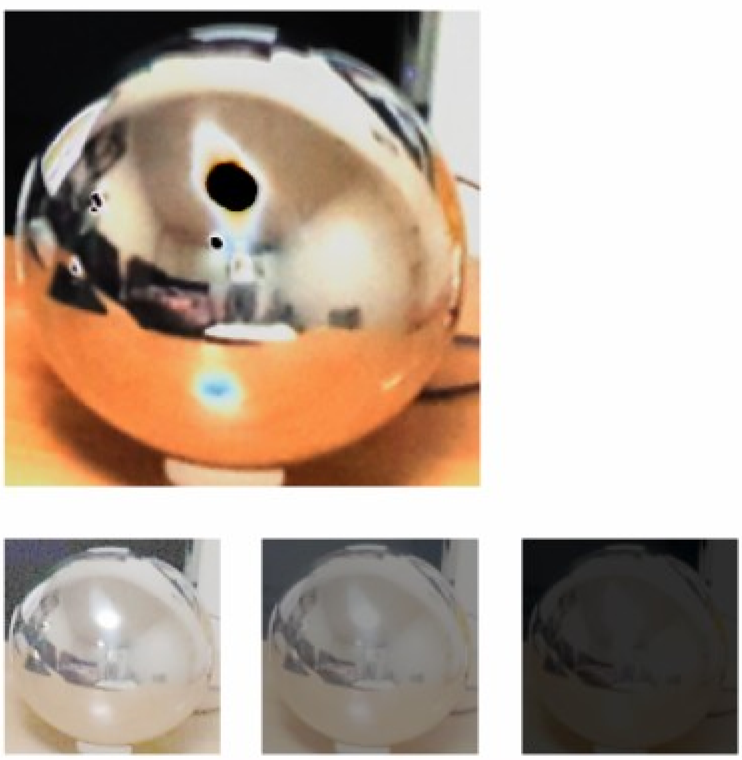
1. Sample data from the starter program

2. My scene

2. LDR merging without under and over exposed regions (10 Points)

In this scenario, we use a weighted averaging procedure as explained in the problem description. The weighted average function is *“w = lambda z: float(128-abs(z-128))”*. Hdr image is calculated by dividing the intensity at a pixel value by the exposure value and multiplying it with a weighted value “w” generated from the above function. Weighted value is generated based on the intensity value at that pixel in all ldr_images. Log irradiance is calculated in pretty much the same way as above mentioned procedure (1).

My Scene



3. LDR merging and response function estimation (15 Points)

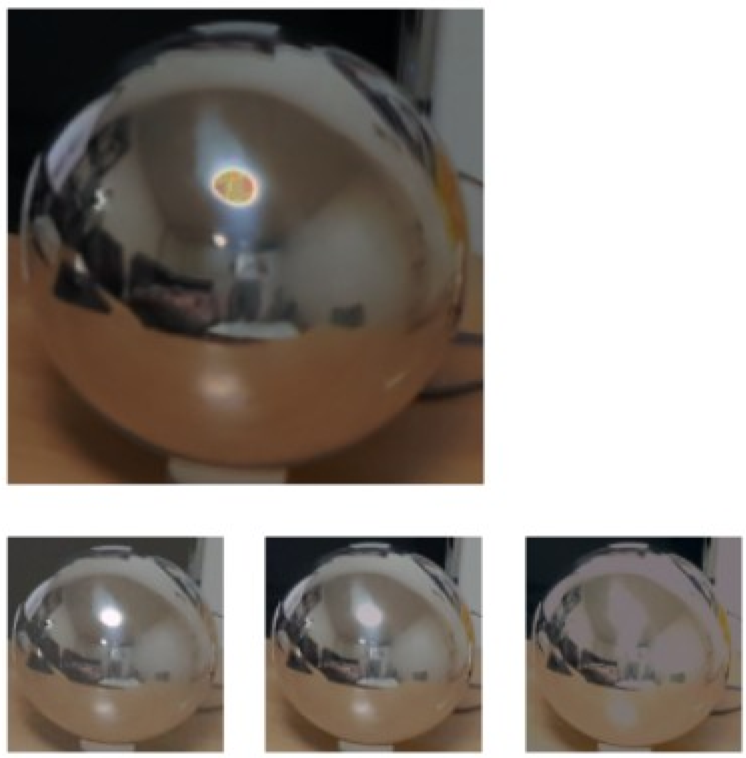
In this scenario, as the steps explained in the problem description, I used a random patch of size 150*150. Below is the pseudo code for the crux of implementation.

```
for each channel
    in every exposure
        for all the random pixels
            get intensity values in Z (N, P)

    solve for g (Z, B, L, W)
    g_per_exposures in each channel is g ??

    # apply the formula
    for all cols (H)
        for all rows (W)
            calculate the formula in each exposure # formula is mention in the paper.
```

Below are the results generated from sample scene and my scene (side-by-side respectively)



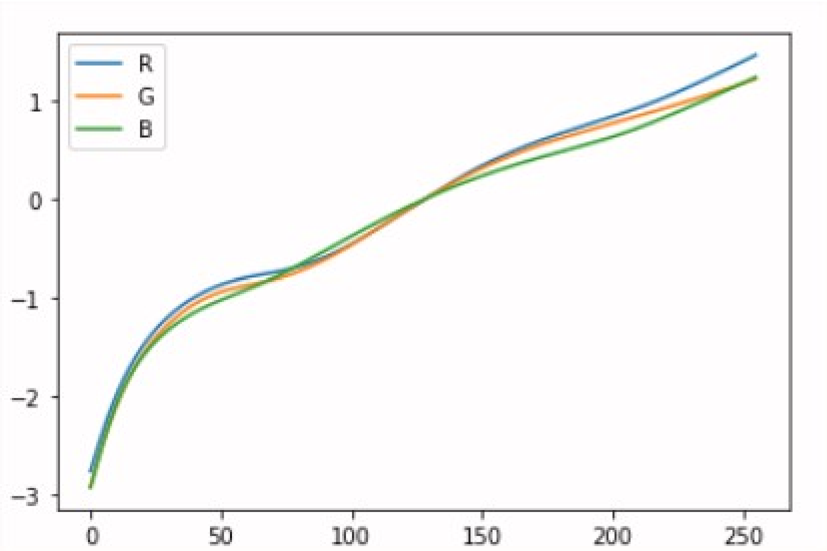
4. Irradiance Discussison

Log radiance is calculated based on the formula *"g(Z) = ln(R*t) = ln(R)+ln(t)"* for naive and filtered apis.

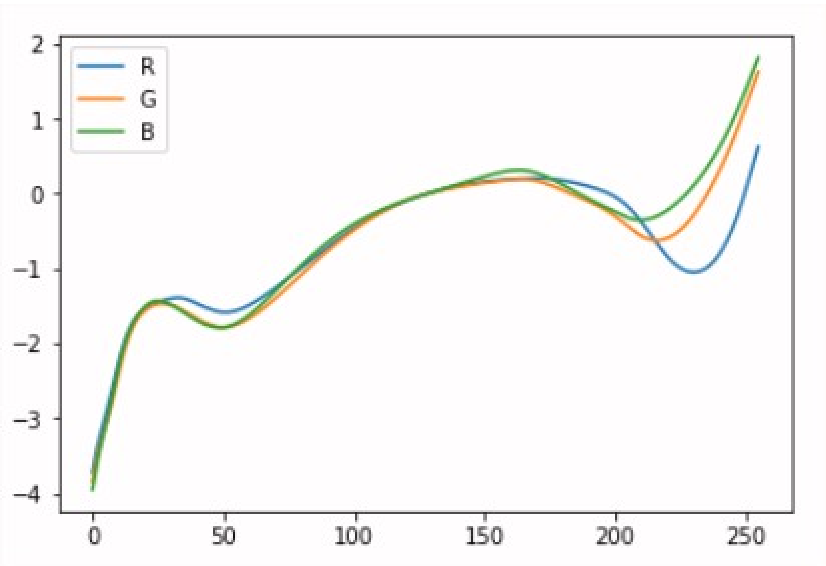
As for estimation, we use the formula from the paper as below.

$$\ln E_i = \frac{\sum_{j=1}^{P} w(Z_{ij})(g(Z_{ij}) - \ln \Delta t_j)}{\sum_{j=1}^{P} w(Z_{ij})}$$

As for the g value graph is plotted as below. As we know based on the formula from the paper, "l" higher the "l" value, better the graph. Below I plotted the graph with a l value of 150 and 25 respectively.



"l" value of 150



"l" value of 25

Panoramic transformations (10 Points)
Panoramic transformation are computed based on the formula “R = V - 2 * dot(V,N) * N,” even though dot(V,N) is not computed. Below is the equation for calculating the equirectangular image “R[:,:,ch] = V[:,:,ch] - 2 * Nz * N[:,:,ch]” for each channels.

Sample scene

My scene

Rendering synthetic objects into photographs (30 Points)

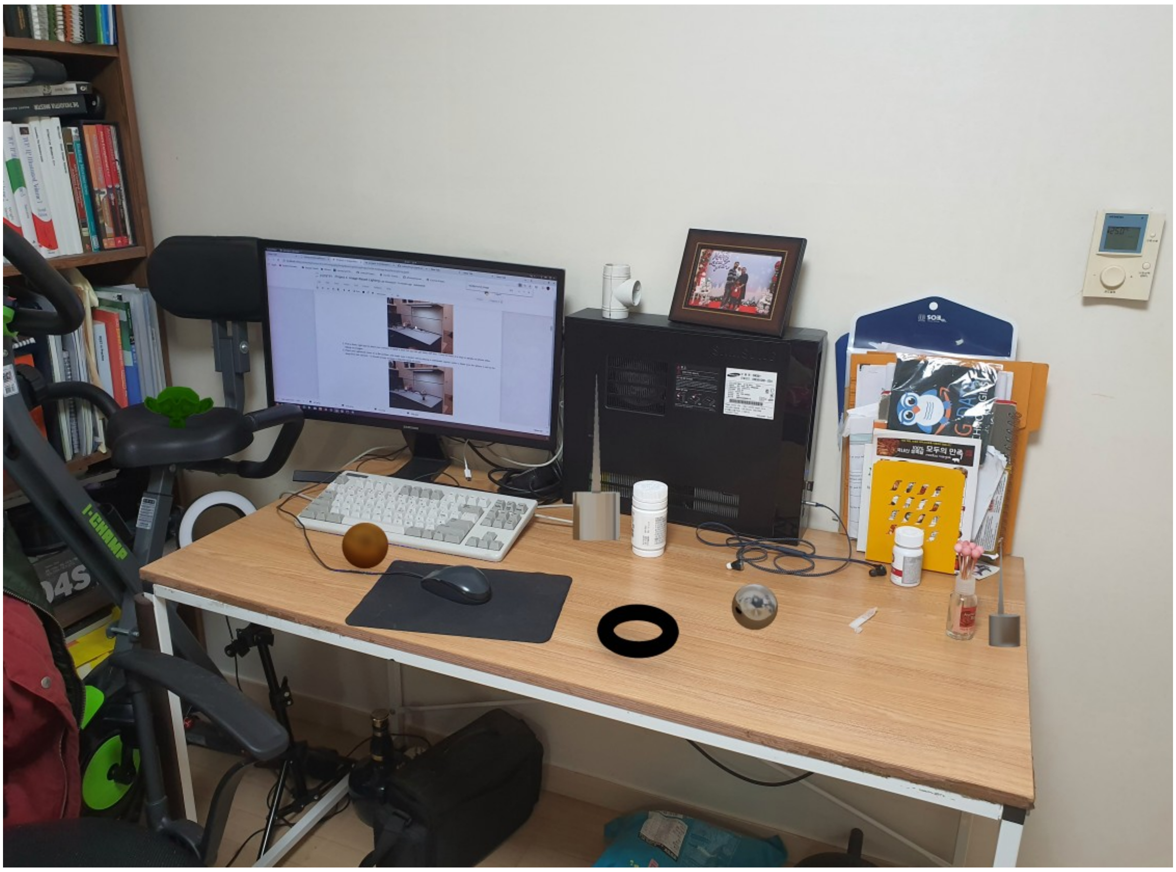
Below are the final blended images for the sample scene and my scene. Blender images "with objects", "without objects" and "mask" can be found in the project folder "samples" and "input/3". Following composite formula is used "composite = M*R + (1-M)*I + (1-M)*(R-E)*c" as mentioned in the problem description. "c" value of .9 is used.

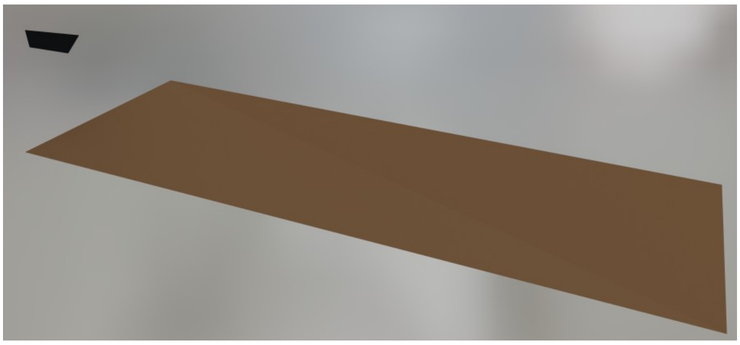
"resized objects from sample data"

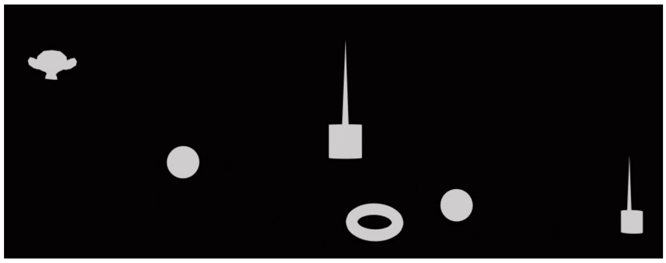
"objects from sample data"

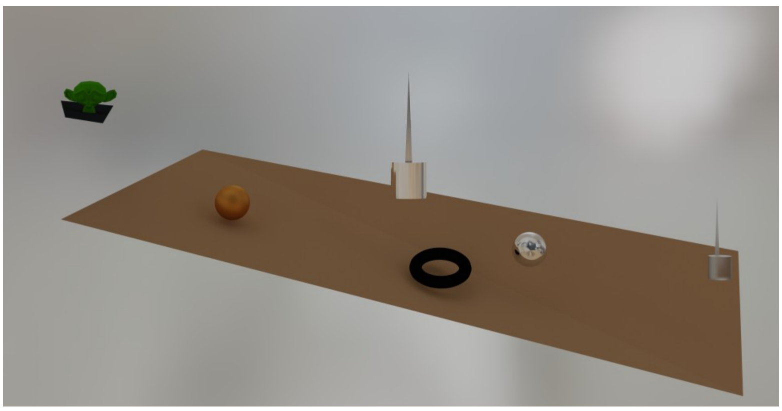
Panoramic Composition from my scene

Below images are self explanatory









Quality of results and Project page

I believe the quality of the results obtained from Sample scene and my scene are worth 100 Points.

Bells & Whistles

Additional Image-Based Lighting Results (20)

As you can see from the below, I have created a new HDR light map and inserted my own objects into it. Below are the snapshots of rendered images with objects, without objects, the mask and the final composite rendered image.