Project 1: High Dynamic Range Imaging

B03902124 黃信元, B03902008 林煦恩

In this project, we have implemented Wards' MTB algorithm for image alignment and compared two HDR creation methods (Debevec's and Robertson's). We have also created a simple ghost removal strategy based on the concept shown in the lecture slides. We use the commercial software Photomatix Pro 5 for tone mapping.

I. Software Usage:

We have written the project using C/C++, and has used openCV for imageIO and linear algebra calculations. Therefore the computer needs to have both openCV and CMake installed.

After installation, typing the following command in the current directory builds up 3 executables: `Debevec_HDR`, `Robertson_HDR` and `MTB` for HDR creation and image alignment.

```
cmake .; make;
```

The usage for `Debevec_HDR` and `Robertson_HDR` are as follows.

```
./Debevec_HDR #Pics threshold <Image_Folder>
./Robertson_HDR #Pics threshold <Image_Folder>
```

<Image_Folder> should includes the the taken photos, named by `img01.jpg`, `img02.jpg` … etc. An metadata named `time.data` should also be included in the folder indicating the time exposure of each images. `#Pics` is the number of pictures in the folder, while `threshold` is the variance threshold used in ghost removal (setting higher threshold creates less ghost region). After execution, an HDR image (in exr format), a picture showing regions with variance above `threshold`, and response curves for RGB (in txt format) are generated.

The usage for `MTB` is as follows.

```
./MTB #Pics <Image_Folder>
```

<Image_Folder> is the folder including `#Pics` pictures. The naming convention should follow as above. This will create a folder named <Image_Folder>_aligned. It contains the images that are aligned and cropped. Then this folder can be used in the above HDR creation. Note that you have to put the metadata containing exposure time in the aligned folder.

II. Taking Photos:

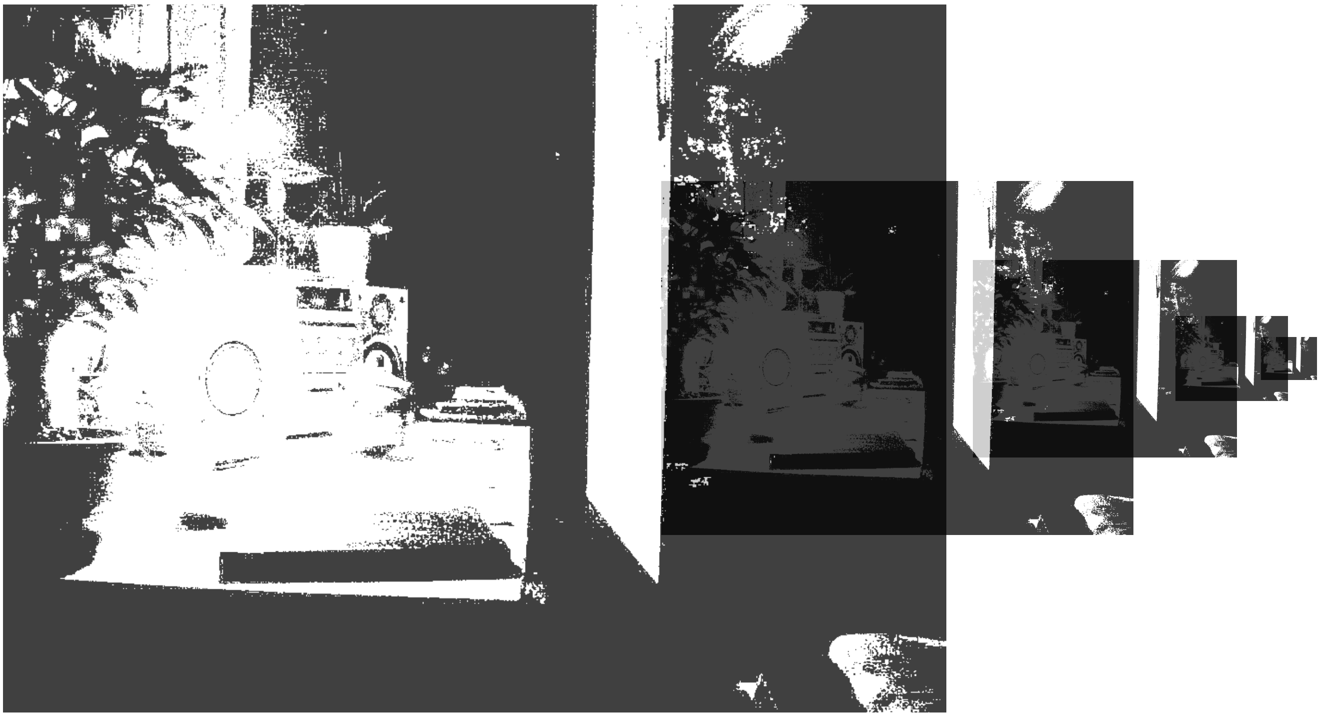
We have used the following cameras in taking photos:

- Lumix DMC-GF7: Lumix DMC-GF7 can be remote controlled using an iPhone app via WiFi. (http://www.panasonic.com/tw/consumer/digital-camera-camcorder/lumix/g-series/dmc-gf7.html)
- Canon 550D: Canon 550D is better than Lumix DMC-GF7 in terms of the optical devices. But can not be remote controlled. (https://www.canon.com.hk/tc/product/catalog/productItemDetails.do?prrfnbr=100365)

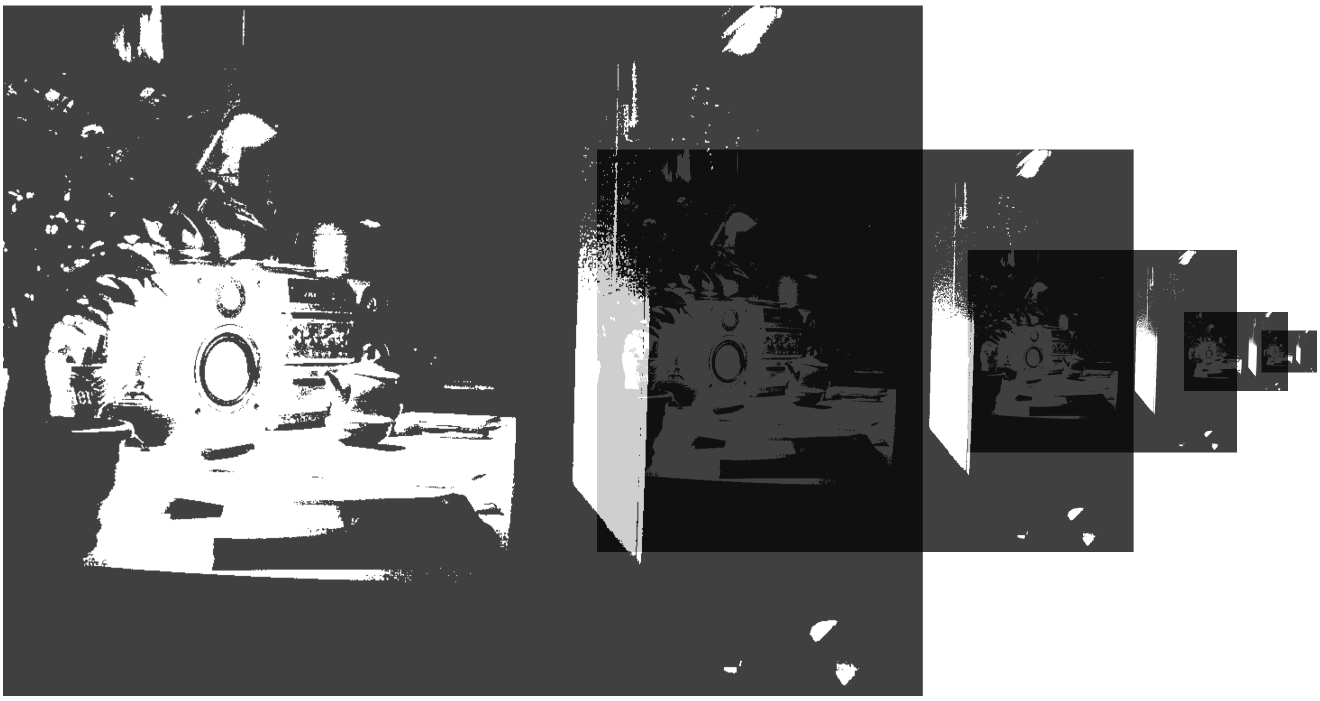
Since they have their own pros and cons, all of them are used in this project. Note that we have to manually change the exposure for Canon 550D, therefore we still implement an image alignment algorithm.

We use the median threshold method introduced in the class to align the images. First of all, we convert each image into grayscale. Then we choose the image with median exposure time as the base image. After that, each image is aligned according to this base image. To calculate the offsets of two images, we first use the median exposure in each picture to threshold the input, and create the binary threshold bitmaps `tb` for each image. We want to minimize the difference of the two bitmaps under some offset between the images. To obtain the optimal offset, we used the pyramid method. This is done by testing 9 offsets from (-1, -1) to (1, 1) in each level and compare the sum of 1's of the two `tbs`' XOR bitmaps. We then choose the offset that minimize the sum. Since there may be some noises in near-median pixels, we first choose a range around median exposure (in this project we take ±4), and then make an excluding bitmap `eb` that will eliminate those untrusted pixels by ANDing it with the `tb`'s XOR bitmap. The process is illustrated in the below figures.

Note that if the base image is chose arbitrarily, the image with extremely high or low exposure time may be picked up and its excluding bitmap can be all 0. If this happened, then all the sum of XOR bitmaps are equal to zero, which may cause the alignment fail. Another small detail that should be noticed is that the default offset should be set to (0, 0). Since the openCV implementation of binary threshold turn all the pixels which has the value less than or equal to the median exposure to 0, once the median exposure equals to 255, all pixels will be set to 0. This will cause an erroneous offset if the default offset is not (0, 0).



This picture shows the thresholded bitmap `tb` for a particular image and the pyramid constructed.



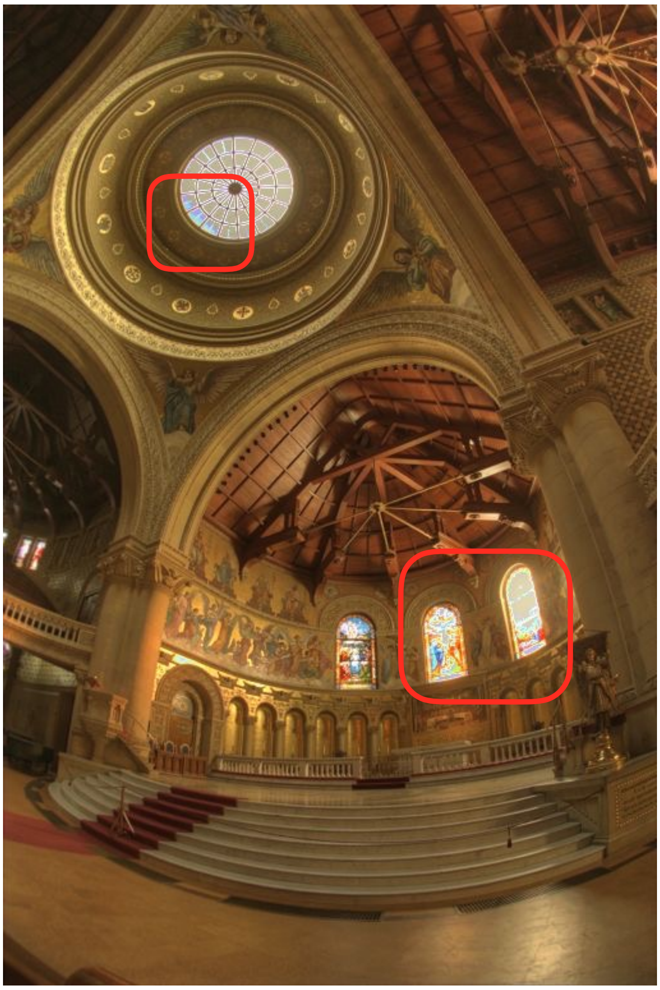
This picture shows the de-noised bitmap `tb&eb` for a particular image and the pyramid constructed. Notice that the bitmaps are less noisy after excluding pixels lying around the median.

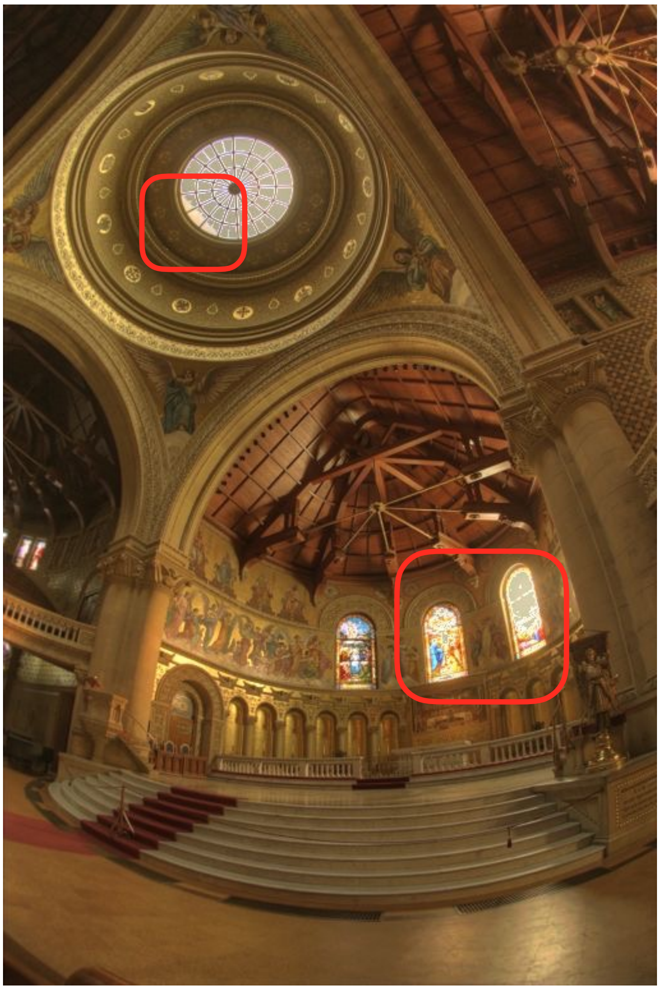
III. HDR Imaging:

We have implemented both Debevec's and Robertson's HDR creation algorithm. This section will summarize the difficulties encountered and the solutions made, additionally we will have some comparisons between the algorithms.

Debevec's HDR algorithm reconstruct the response function by selecting a set of locations on the image. Instead of hand picking the locations, we want an automatic scheme. The simplest method is to random sample a fixed number of points. However this may not be evenly spread across the image. Therefore we use a grid with roughly 70 points evenly spread on the image to be our sampled points (although deterministically). Then we proceed with the usual process of constructing the matrix, and solving the least square problem. Since we are using C/C++, some linear algebra library has to be used to solve the least square problem. Luckily we found that the Image I/O library OpenCV also includes an API to do pseudo-inverse for a given matrix. Thus we make use of this convenient function. For the response curve, each channel is conducted independently. Thus we also calculate the final irradiance by the formula shown to the left also independently for each channel. However, it may create the following image to the left. *(Note that for all the memorial images, we only use the 8 pictures with the highest exposure, since they have minimal amount of pure blue color, and some issues can be more clearly shown)* Both images are tone-mapped using Photomatix's default option. It can be clearly seen to have some weird blue shift in the image. I suspect that it is because different exposures are weighted differently in different channels. Thus I have consider using the same weighting for different exposures in different channels. This is done by averaging the weights in the 3 channels RGB. The final image is shown to the right. It did not exhibits the blue shift, however the image is a little broken since only 8 exposures are used.

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





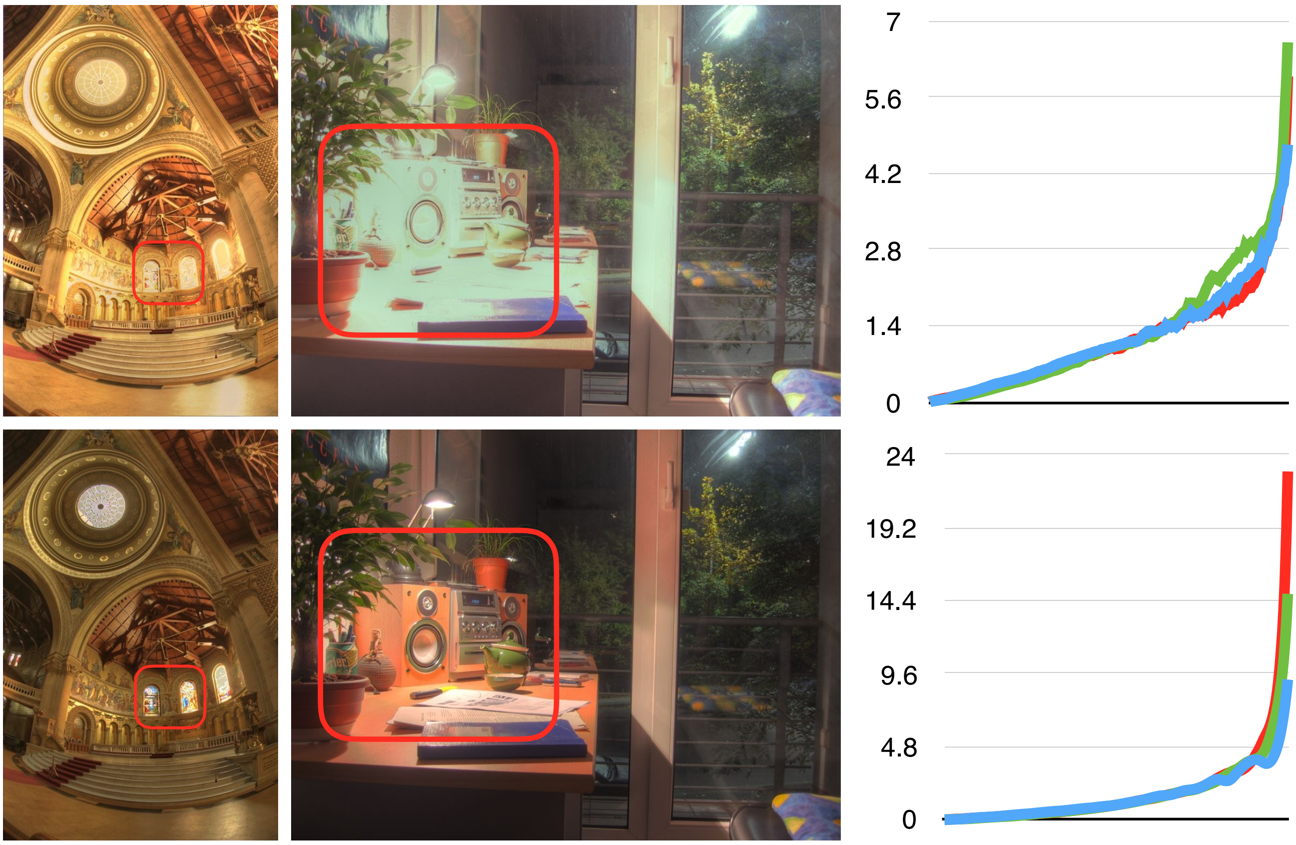
Robertson's HDR algorithm uses a Gaussian weighting function instead of the triangle weight as in Debevec's algorithm. And it optimize a different objective function, which is shown to the left. In my implementation, different channels are considered independently (Nevertheless I did not observe similar effects as in Debevec's HDR algorithm). And all pixels are used in the optimization.

$$\tilde{O}(\mathbf{I}, \mathbf{x}) = \sum_{i,j} w_{ij} \left(I_{y_{ij}} - t_i x_j \right)^2$$

Thus solving the optimization problem will also determine the final high dynamic image (However we have implemented ghost removal technique, so the situation is more complex). To solve the above optimization, we use the standard alternating least square (ALS) algorithm with 30 complete iterations. The initial condition for the response function is simply selected as a linear function i.e. I(x) = x / 128, with x ranging from 0 to 255. A problem that I have discovered about Robertson's HDR is that it depends too much on high exposure pictures. This can actually be seen theoretically from the formula used to calculate a pixel's irradiance value (shown above). By a simple rearrangement, we can arrive at the equation shown to the right. We can clearly see that the exposure time term actually work as a weighting function. And a picture with 16 times of exposure is weighted 256 times more. This may give rise to severe break down due to the whiteout in high exposure pictures. To verify my finding, we compare the memorial and radio image using Debevec's and Robertson's algorithm. Both tone-mapped by Photomatix's default option. The upper images are by Roberson's algorithm, while the bottom images are by Debevec's algorithm. Roberson's algorithm is much whiter with many details not clearly shown. And the circled region is completely broken due to large bias to high exposure pictures. Therefore we think Debevec's algorithm is more suitable and will use Debevec's algorithm for subsequent experiments. The response curve for radio image is also shown accordingly, where Debevec's algorithm yields smoother curves due to the regularization term.

$$E_i = \frac{\sum_j w(Z_{ij}) g(Z_{ij}) \Delta t_j}{\sum_j w(Z_{ij}) \Delta t_j^2}$$

$$E_i = \frac{\sum_j w(Z_{ij}) E_{ij} \Delta t_j^2}{\sum_j w(Z_{ij}) \Delta t_j^2}, \text{If } E_{ij} \equiv g(Z_{ij}) / \Delta t_j$$



IV. Ghost Removal:

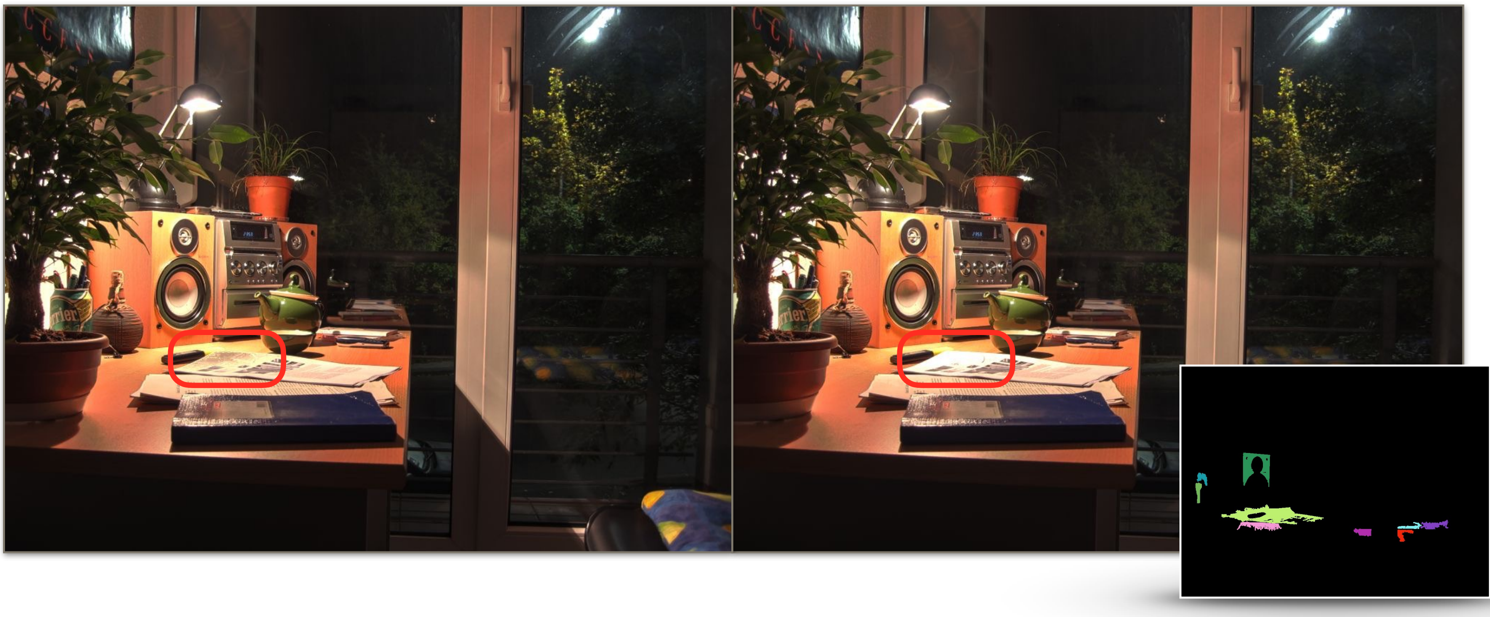
Since there are many scenes with some occasionally moving objects, it is important to do ghost removal. We have also discover ghost removal strategy to work great when there are too much whiteout (e.g. memorial image with 8 pictures).

The idea of our implementation is based on the concept shown in the lecture slide. First of all, the variance of each pixels is calculated. This is done using the normalized variance shown to the right. Then, pixels with variance higher than some threshold is extracted. Later, we dilate each high variance pixels into a 3x3 pixel block. Blocks that intersect with each other are group together forming regions. In our implementation we easily achieve this using disjoint set (union-find tree). After then, large regions are selected (cover at least 0.1% of the image). For each of the large region (ghost region), we need to find the best exposure. We did this by calculating the sum of the weighting function (triangle weight) in each region for a given exposure. The exposure with the maximum sum is defined as the best exposure. The intuition is that when maximizing the sum of weights, the pixels will mostly lie around 128, which should has the best resolution. Finally, to create the HDR image, we apply a weighted average of the original irradiance and the irradiance in the best exposure. Note that for pixels not within any ghost region, it simply use the original irradiance. The weighted average is as follows. This completes our ghost removal process.

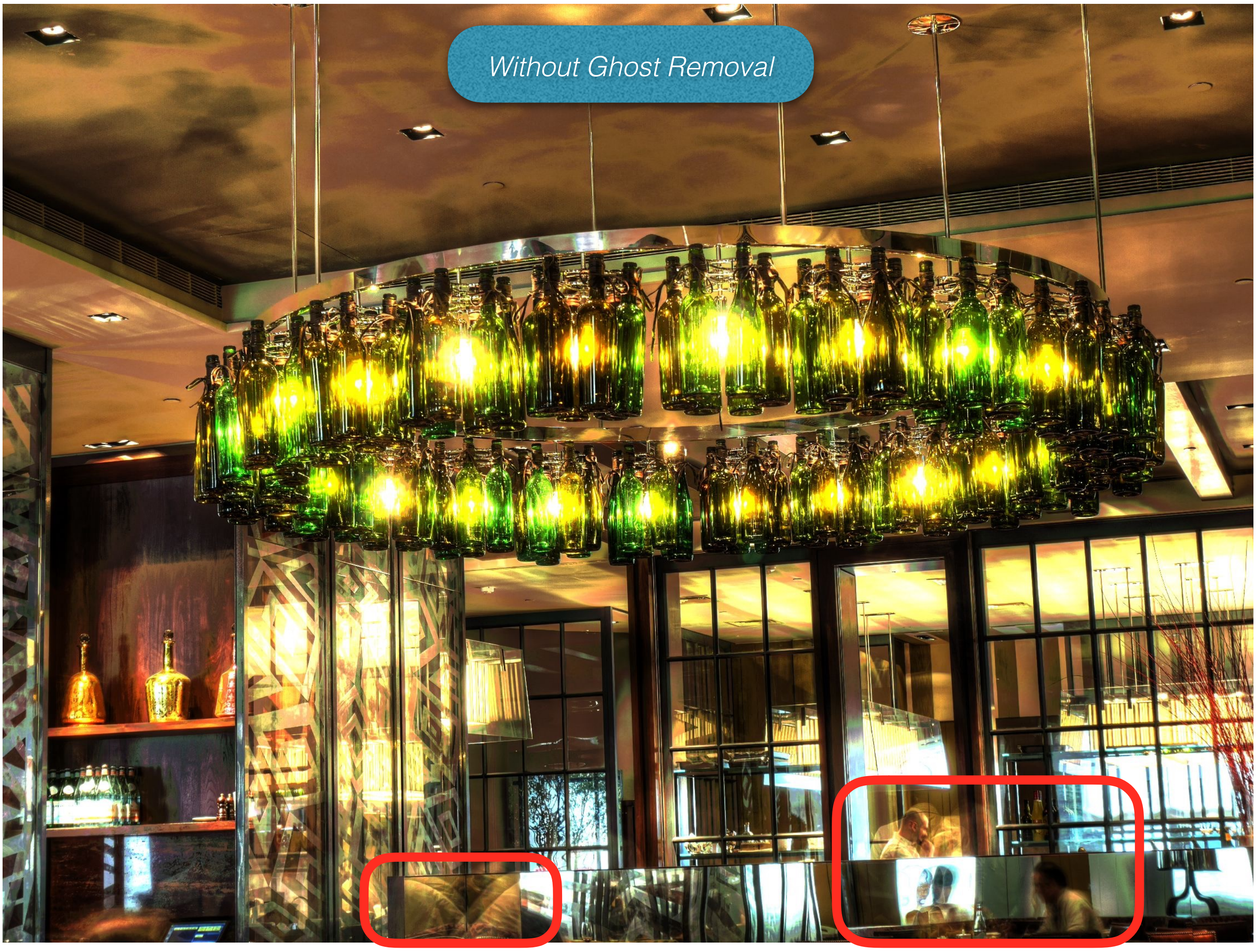
$$\frac{(\sum_j w(Z_{ij})E_{ij}^2)/(\sum_j w(Z_{ij}))}{(\sum_j w(Z_{ij})E_{ij})^2/(\sum_j w(Z_{ij}))^2} - 1$$

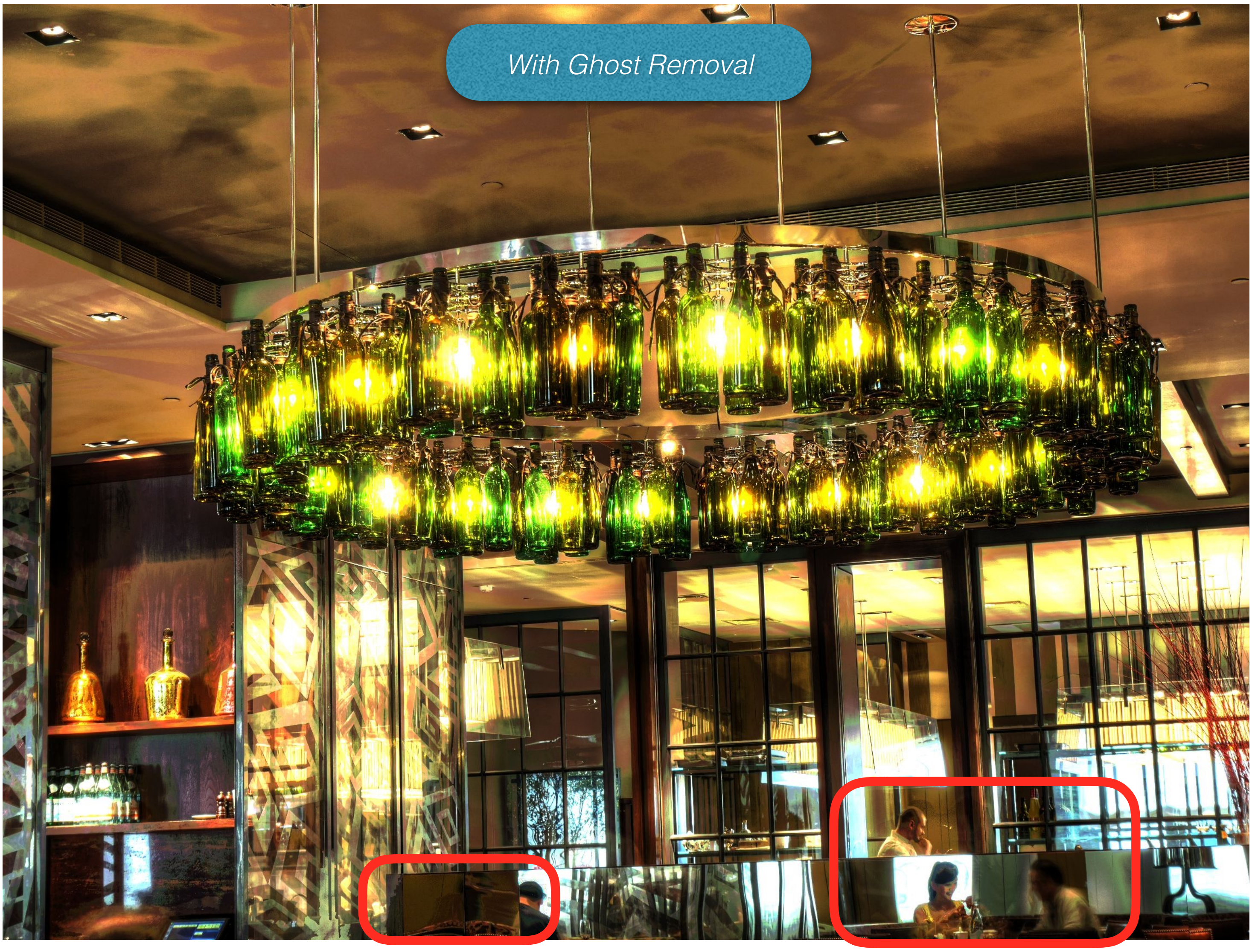
$$\alpha * \text{best\_expo} + (1-\alpha) * \text{original}, \text{where } \alpha = min(\frac{\text{Variance}}{\text{Threshold}}, 1)$$

To show the effectiveness of the simple ghost removal strategy, we again use the memorial image with 8 highest exposures and tone-mapped by Photomatix default option. This image did not contain any ghost, but there are many whiteout on the windows. The whiteouts can be corrected by this simple ghost removal strategy as shown in the following comparison. The middle image did not apply our ghost removal, while the rightmost image has apply the ghost removal strategy. The leftmost image is the regions detected to have high variance. We can clearly see that the reconstructed HDR image is much better with ghost removal applied. Another example is the radio image with 8 highest exposures and tone-mapped by Photomatix enhanced 2. Image without ghost removal is shown to the left, while image with ghost removal is shown to the right. The bottom right picture shows the regions detected to have high variance. (Threshold set to 0.3)



All of the above image did not contain ghost, however our method still works great in improving the final image. (This may be because the simple ghost removal strategy works some how like exposure fusion in certain areas) To show the effectiveness in removing ghost, we use the following pictures taken in Italian restaurant Bencotto in The Arcade for HDR creation. The high variance regions are shown in the bottom right picture. (Threshold set to 0.25) And the HDR image is tone mapped using Photomatix “Surreal” option (with light adjustment to “Natural”). The final image with ghost removal is much better than that without ghost removal (as shown in the circled area). However, some of the objects are still blurry. This is because the objects in the original pictures are already blurred due to high exposure time (a few seconds). We conclude that our simple ghost removal strategy works pretty good.


Without Ghost Removal


With Ghost Removal

V. HDR Photo Gallery:

In this section, we present our tone-mapped photos. It is created from the HDR image using Photomatix's detail enhancer (with hand-tuned parameter). The original pictures are in the folders under this directory. The generated HDR images and the tone-mapped images are in the corresponding folders (Due to upload file size limitation, the generated HDR image for Clock and Arcade are not included in their directory, please contact me to obtain them or create on your own via the above instructions). The image showing high variance region is also in the same folder.

Fig 1. Clock: Taken using Canon 550D, tone-mapping modified from Photomatix's "painterly 3" option, variance threshold set to 0.065.

Fig 2. Arcade: Taken using Canon 550D, tone-mapping modified from Photomatix's "painterly 4" option, variance threshold set to 0.15.

Fig 3. Bottles: Taken using Lumix DMC-GF7, tone-mapping modified from Photomatix's "surreal" option, variance threshold set to 0.25.

Fig 4. Flower: Taken using Lumix DMC-GF7, tone-mapping modified from Photomatix's "painterly 4" option, variance threshold set to 0.2.

Fig 5. Corridor: Taken using Lumix DMC-GF7, tone-mapping modified from Photomatix's "painterly 4" option, variance threshold set to 0.1.

Fig 6. Wine: Taken using Lumix DMC-GF7, tone-mapping modified from Photomatix's "painterly 4" option, variance threshold set to 0.5.

VI. Reference:

[1] Greg Ward, Fast Robust Image Registration for Compositing High Dynamic Range Photographs from Hand-Held Exposures, Journal of Graphics Tools 2003.

[2] Paul E. Debevec, Jitendra Malik, Recovering High Dynamic Range Radiance Maps from Photographs, SIGGRAPH 1997.

[3] Mark Robertson, Sean Borman, Robert Stevenson, Estimation-Theoretic Approach to Dynamic Range Enhancement using Multiple Exposures, Journal of Electronic Imaging 2003.

[4] Reinhard, Erik and Ward, Greg and Pattanaik, Sumanta and Debevec, Paul, High Dynamic Range Imaging: Acquisition, Display, and Image-Based Lighting.