**PBI Report**

**PDPM IIITDM Jabalpur**

**Internship at Canon India Pvt. Ltd**

**Topic - High Dynamic Range (HDR) imaging and Tone mapping algorithms**

**External Mentor – Radhish Ayyappan**

**Internship Duration – 16th May to 15th November**

**Report for Duration – 16th May to 15th June**

|  |  |
| --- | --- |
| **From** | **To** |
| **Ravi Jain** | **Dr. Ayan Seal** |
| **B.Tech**  **Computer Science and Engineering** | **Assistant Professor Computer Science and Engineering** |
| **2013167** |  |
| **2013167@iiitdmj.ac.in**  **ravithats.it@rediffmail.com** | **ayan@iiitdmj.ac.in** |

**HDR Image Generation –**

This report includes the steps involved in understanding tone mapping algorithms and comparing the results of each algorithm.

Tone mapping methods provided by **pfstools** are -

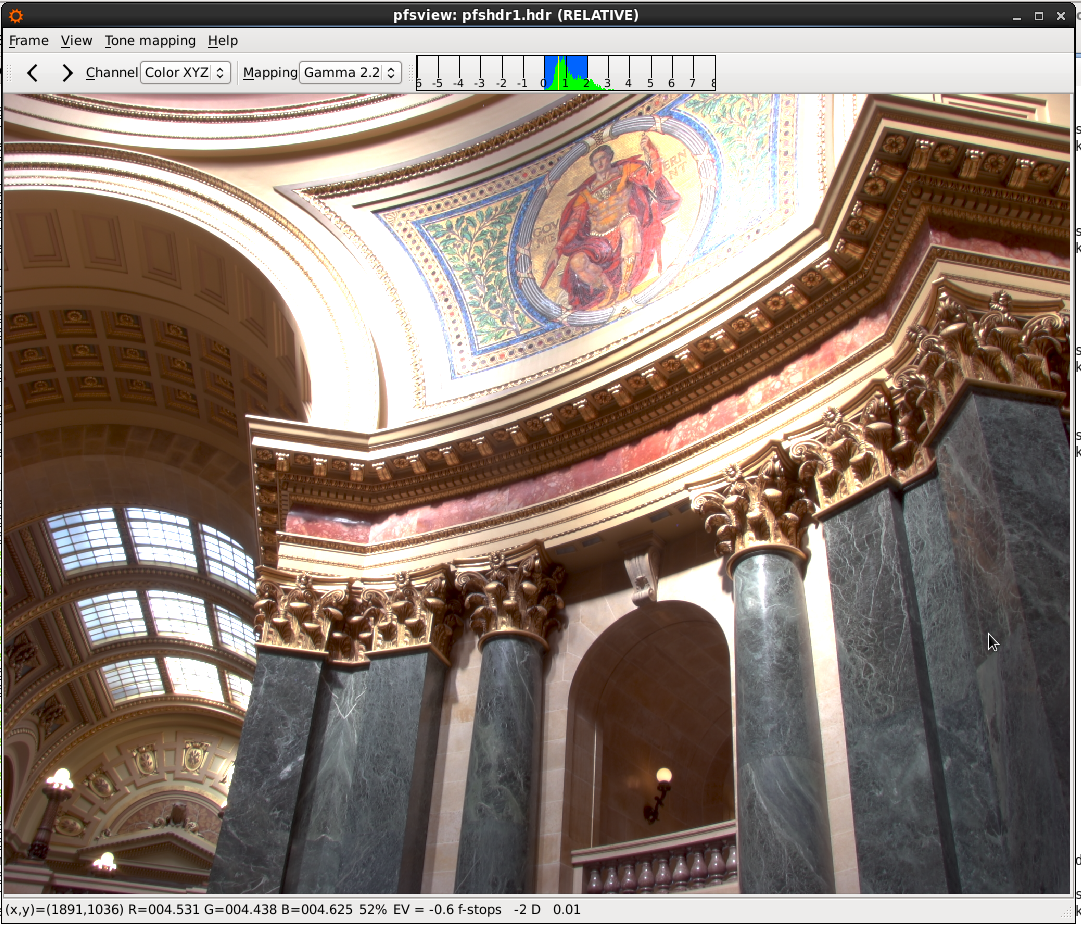
* [**pfstmo\_drago03**](http://pfstools.sourceforge.net/man1/pfstmo_drago03.1.html) - adaptive logarithmic tone mapping operator
* [**pfstmo\_durand02**](http://pfstools.sourceforge.net/man1/pfstmo_durand02.1.html) - fast bilateral filtering tone mapping operator
* [**pfstmo\_fattal02**](http://pfstools.sourceforge.net/man1/pfstmo_fattal02.1.html)- gradient domain tone mapping operator
* [**pfstmo\_mantiuk06**](http://pfstools.sourceforge.net/man1/pfstmo_mantiuk06.1.html) - contrast domain tone mapping operator
* [**pfstmo\_mantiuk08**](http://pfstools.sourceforge.net/man1/pfstmo_mantiuk08.1.html) - display adaptive tone mapping operator
* [**pfstmo\_pattanaik00**](http://pfstools.sourceforge.net/man1/pfstmo_pattanaik00.1.html) - time-dependent visual adaptation tone mapping operator
* [**pfstmo\_reinhard02**](http://pfstools.sourceforge.net/man1/pfstmo_reinhard02.1.html) - photographic tone mapping operator
* [**pfstmo\_reinhard05**](http://pfstools.sourceforge.net/man1/pfstmo_reinhard05.1.html) - photoreceptor tone mapping operator
* [**pfstmo\_mai11**](http://pfstools.sourceforge.net/man1/pfstmo_mai11.1.html)- tone-mapping for backward-compatible compression
* [**pfstmo\_ferradans11**](http://pfstools.sourceforge.net/man1/pfstmo_ferradans11.1.html) - tone mapping operator with visual adaptation and local contrast enhancement

We try to compare the output of these tone mapping algorithms and the time taken by them to tone map a particular HDR image.

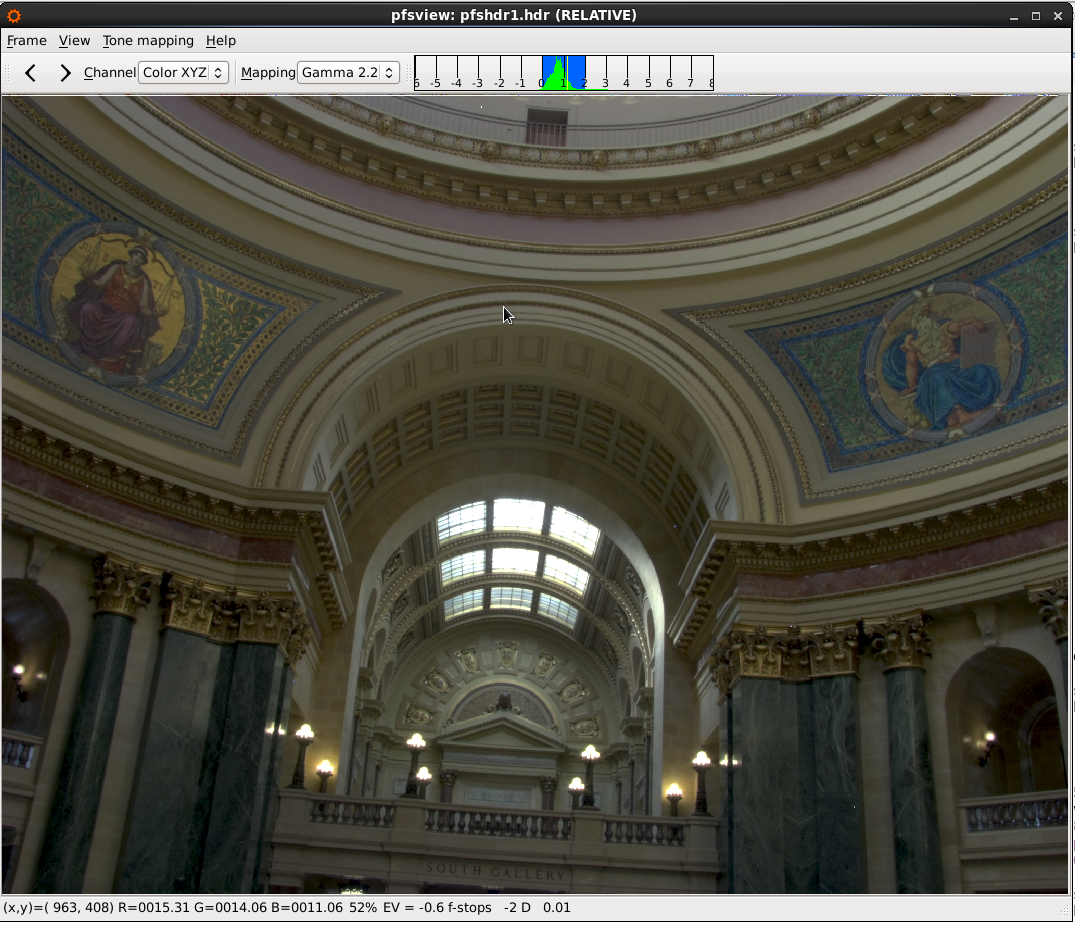
Below are the HDR images we have used to compare these algorithms.

**Sample HDR Images -**

**Madison Capitol - 1**



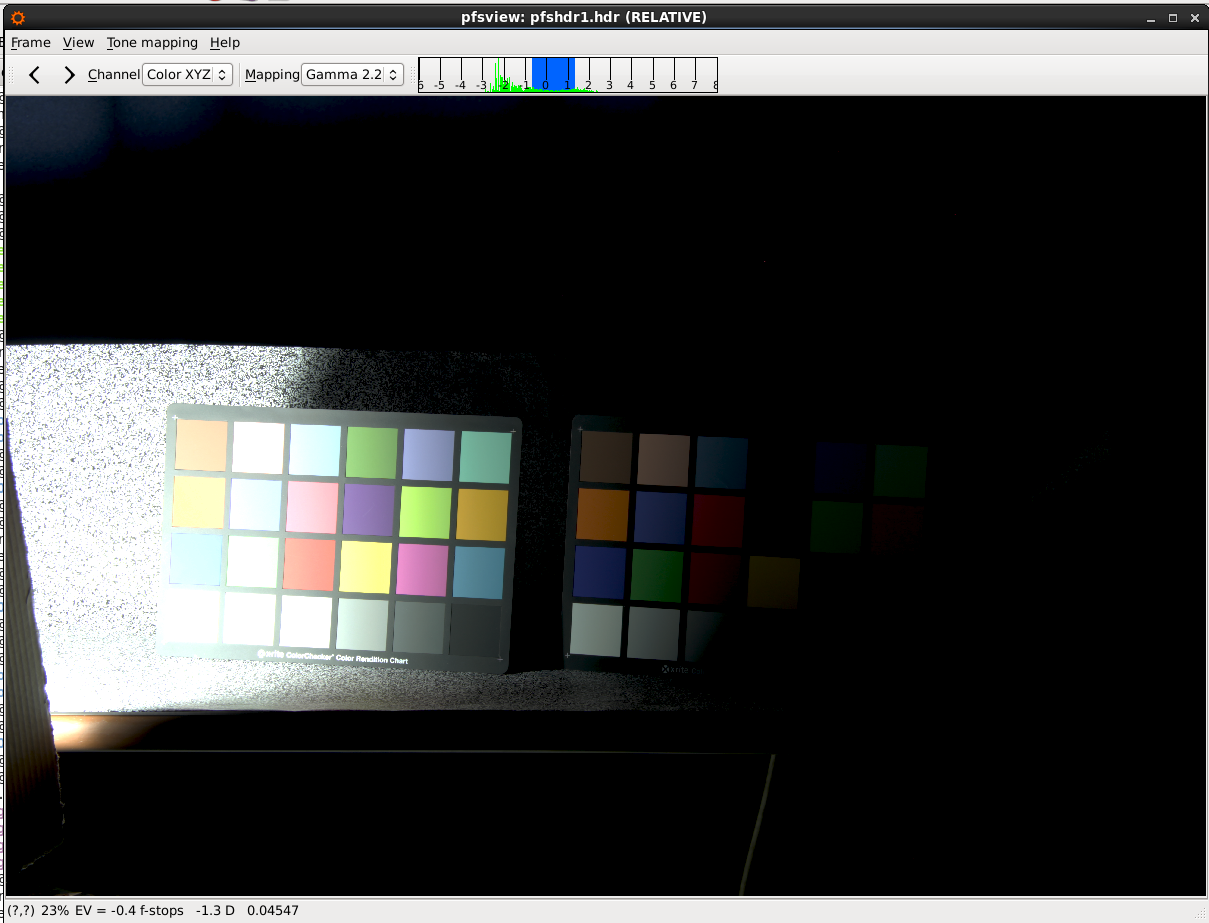
**Madison Capitol - 2**



**Shuttle HDR -**



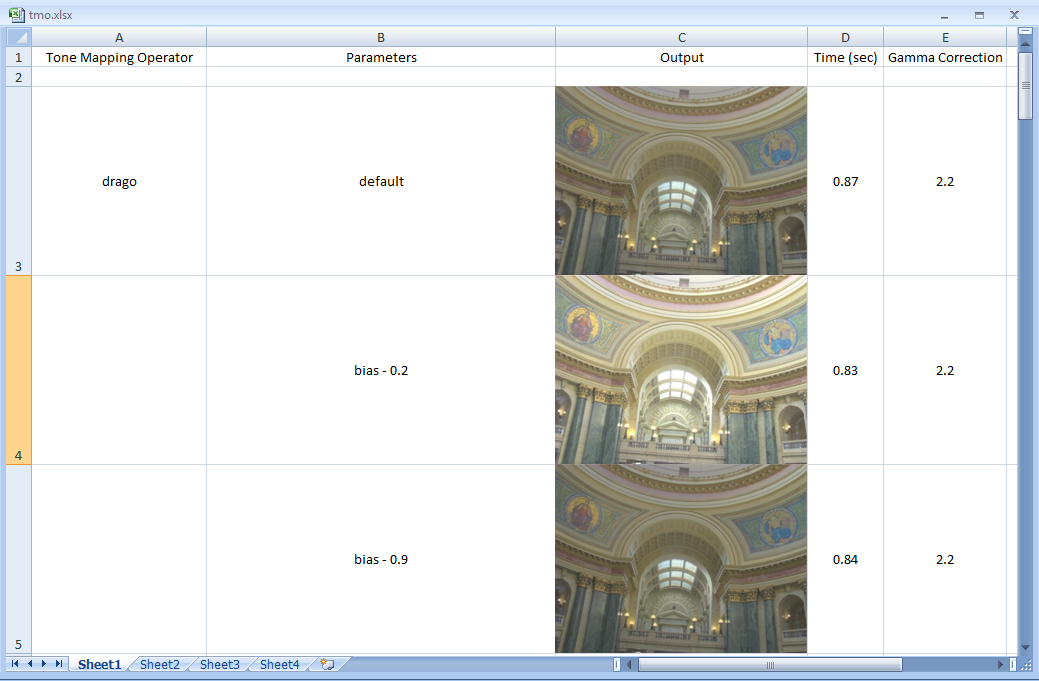
**Macbeth Colour Checker -**



**Procedure –**

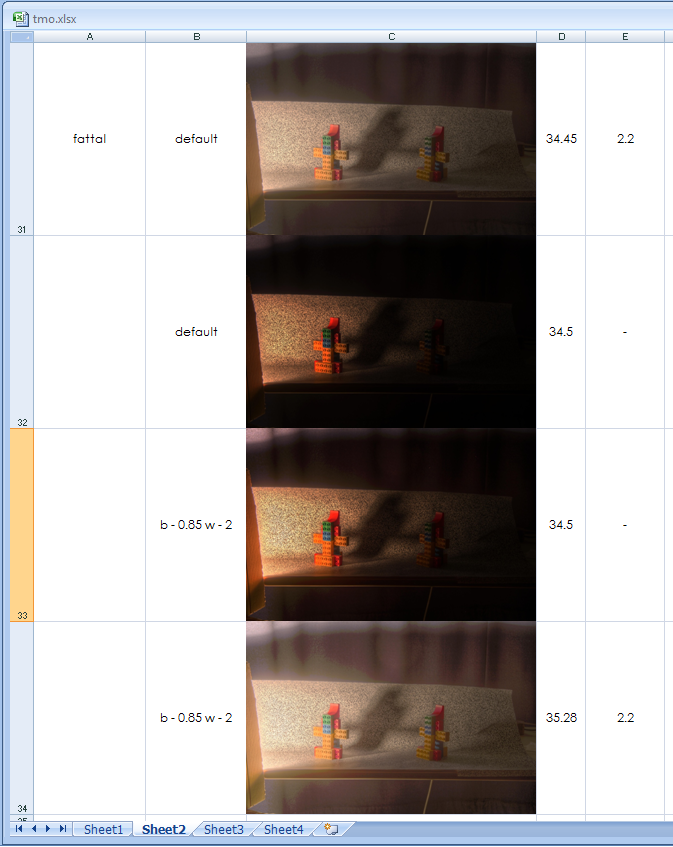
1. Take an HDR image as input.
2. Apply the tone mapping operators mentioned above one by one.
3. Evaluate the time taken by each in order to achieve the final tone mapped image (for this we had to modify the pfstools code in order to display the time taken in various stages of an algorithm).
4. Change the parameters of each algorithm, for example gamma correction value, bias, brightness, contrast, saturation etc.
5. Note the readings, save the output images in an excel spreadsheet and draw comparison between these tone mapping algorithms.

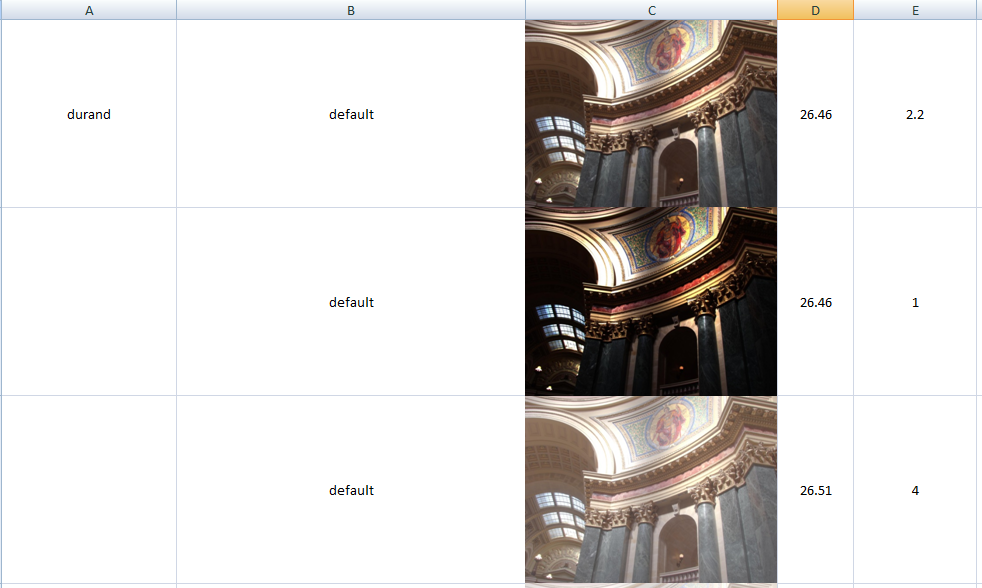
**Excel workbook Screenshots –**



In the above screenshot we display the output obtained by applying Durand tone mapping operator on Madison Capitol -2.

The time taken and gamma correction values are mentioned along with the parameters (bias value).







The complete excel workbook is uploaded at . Kindly refer to the four sheets in the workbook for more details.

**Observations –**

* Durand and Ferradans operators are more time consuming in comparison to others, so there is scope for reducing time complexities in these algorithms.
* Pattanaik operator fails to show good tone mapped images when the input is a high definition HDR image (shuttle HDR resolution is 5184x3456).
* On modifying a few parameters mantiuk operator fails to produce a tone mapped output for high resolution HDR image.
* Ferradans operator requires much higher RAM than others, and the machine which we used failed to use this operator for higher resolution HDR image.

**Tools Used**

* pfstools
* ImageMagick
* QT viewer for displaying hdr images
* OpenGL
* MS Excel

**Conclusion –**

* The readings and output were successfully entered into an excel workbook.
* Time taken by each operator was evaluated.
* Certain observations were made regarding these operators.
* **The next step is to parallelise these algorithms and find more bugs in the software, so that all HDR images can be successfully tone mapped.**

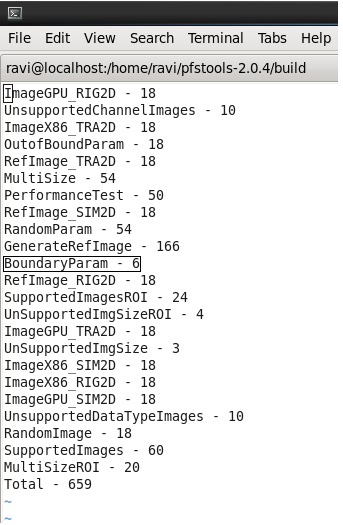
**Python Programming Task**

The task given in the first week to me was extended and I had to modify its code. The program now would –

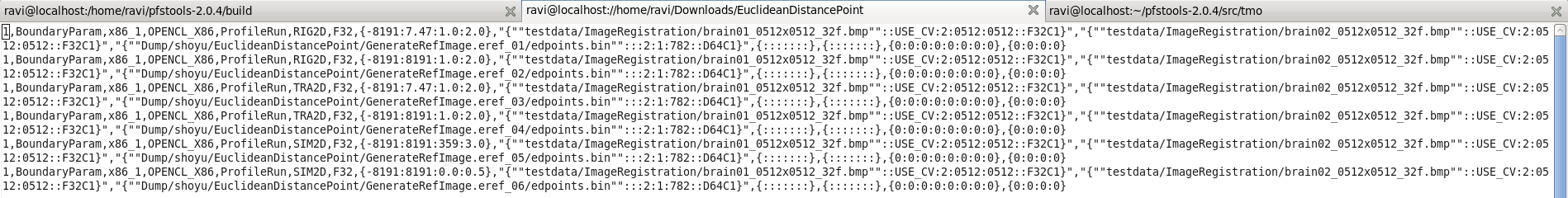
* Take a csv (comma separated values) file as input that contains information related to different scenarios.
* Create separate csv file for each scenario.
* Create another file which stores the count of each scenario (along with scenario name) in the original csv file.
* Sum up this count and display the total number of tuples in the csv file.

**The Screenshots of the output obtained are –**

**Output text file -**



**Output csv file -**



**We see there are 6 tuples in BoundaryParam scenario**

**Tools Used**

* Python-3.5.1
* gedit

**References**

* http://pfstools.sourceforge.net/index.html
* http://pages.cs.wisc.edu/~csverma/CS766\_09/HDRI/hdr.html
* http://eidomatica.di.unimi.it/index.php/research/idb/yaccd2
* http://docs.opencv.org/3.0-beta/doc/tutorials/photo/hdr\_imaging/hdr\_imaging.html
* http://www.stackoverflow.com
* https://docs.python.org/3/library/csv.html