HDR Imaging and Tone Mapping Algorithms

by

Ravi Jain

(Roll No. 2013167)

Supervisor (s) :

External :   **Internal :**

Mr. Radhish Ayyappan   Dr. Ayan Seal

Project Manager   Assistant Professor

Canon ISDC, Bangalore   PDPM IIIT DM Jabalpur

Computer Science & Engineering (B.Tech 2013)

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY, DESIGN AND MANUFACTURING JABALPUR

3rd Report

(12th July 2016 - 26th July 2016)

Due to near release date of our project in the company, I had to work on designing a dashboard using python language. The HDR imaging topic will continue after the release of this project.

Introduction –

- The aim of this project is to design a dashboard for certain csv(comma separated value) and log files, which would display details of these in addition to certain graphs for indicating kernel execution times.
- The python program takes as input a csv file, its corresponding log file, median value files (if any) and path to the images which can be viewed in the dashboard.
- The skills used for the development of this dashboard are python, angularjs, javascript, bootstrap, css, html, matplotlib and jquery.

Procedure -

The dashboard serves the purpose of displaying details such as kernel execution time, total time etc. on executing certain algorithms over several images and drawing a comparison between different test environment types.

The set of csv files include details of algorithms such as sum of squared differences, mutual induction, gradient image filter etc.

The aim was to display the logs of execution of these algorithms over certain images.

Furthermore, graphs were also made using matplotlib, including comparison plots, box plots. Modules such as numpy were used.

The initial task was to understand use of these tools and install libraries on my system.

Sample Files –

Csv file -

Log file –

Median log file –

```
C:\Users\alkaggc\AppData\Local\Temp\Temp1_chartPlot1.zip\performance_VecOperation_CPU_logs_800 - Notepad++
File  Edit  Search  View  Encoding  Language  Settings  Macro  Run  Plugins  Window  ?

1588  VectorVectorAdd_X86_I16_0512x0512x2,    2.91211,    8.53276,    2.10791,   13.55278,   0.00000,   0.00000,   0,   0.00000,
1589  VectorVectorAdd_X86_I16_0512x0512x2,    2.77490,    6.18426,    0.83203,    9.79119,   0.00000,   0.00000,   0,   0.00000,
1590  VectorVectorAdd_X86_I16_0512x0512x2,    2.91284,    7.59090,    2.21680,   12.72054,   0.00000,   0.00000,   0,   0.00000,
1591  VectorVectorAdd_X86_I16_0512x0512x2,    3.14111,    2.33186,    0.84912,    6.32209,   0.00000,   0.00000,   0,   0.00000,
1592  VectorVectorAdd_X86_I16_0512x0512x2,    3.01294,    5.67626,    5.30908,   13.99828,   0.00000,   0.00000,   0,   0.00000,
1593  VectorVectorAdd_X86_I16_0512x0512x2,    3.40308,    3.43245,    0.87402,    7.70955,   0.00000,   0.00000,   0,   0.00000,
1594  VectorVectorAdd_X86_I16_0512x0512x2,    2.92407,    5.75588,    0.88916,    9.56911,   0.00000,   0.00000,   0,   0.00000,
1595  VectorVectorAdd_X86_I16_0512x0512x2,    3.09082,    2.53510,    0.83887,    6.46479,   0.00000,   0.00000,   0,   0.00000,
1596  VectorVectorAdd_X86_I16_0512x0512x2,    2.82617,    2.36075,    0.67090,    5.85782,   0.00000,   0.00000,   0,   0.00000,
1597  VectorVectorAdd_X86_I16_0512x0512x2,    2.52588,    8.22904,    0.64404,   11.39897,   0.00000,   0.00000,   0,   0.00000,
1598  VectorVectorAdd_X86_I16_0512x0512x2,    2.07593,    3.00422,    0.82812,    5.90827,   0.00000,   0.00000,   0,   0.00000,
1599  VectorVectorAdd_X86_I16_0512x0512x2,    1.96802,    4.46070,    1.04688,    7.47559,   0.00000,   0.00000,   0,   0.00000,
1600  VectorVectorAdd_X86_I16_0512x0512x2,    3.46606,    3.70699,    0.89087,    8.06392,   0.00000,   0.00000,   0,   0.00000,
1601  VectorVectorAdd_X86_I16_0512x0512x2,    3.66211,    2.95753,    0.84009,    7.45972,   0.00000,   0.00000,   0,   0.00000,
1602  VectorVectorAdd_X86_I16_0512x0512x2,    3.13184,    8.31008,    0.79712,   12.23903,   0.00000,   0.00000,   0,   0.00000,
1603  VectorVectorAdd_X86_I16_0512x0512x2,    2.91797,    8.78260,    0.66089,   12.36145,   0.00000,   0.00000,   0,   0.00000,
1604  VectorVectorAdd_X86_I16_0512x0512x2,    2.58276,    7.47740,    0.75879,   10.81895,   0.00000,   0.00000,   0,   0.00000,
1605  VectorVectorAdd_X86_I16_0512x0512x2,    2.01196,    8.38417,    0.67798,   11.07412,   0.00000,   0.00000,   0,   0.00000,
1606  VectorVectorAdd_X86_I16_0512x0512x2,   10.48096,    4.51508,    0.77808,   15.77411,   0.00000,   0.00000,   0,   0.00000,
1607  VectorVectorAdd_X86_I16_0512x0512x2,    2.74902,    6.52209,    1.32910,   10.60022,   0.00000,   0.00000,   0,   0.00000,
1608  VectorVectorAdd_X86_I16_0512x0512x2,    2.93799,    3.30095,    0.66309,    6.90203,   0.00000,   0.00000,   0,   0.00000,
1609  VectorVectorAdd_X86_I16_0512x0512x2,    2.04004,    5.72355,    0.76807,    8.53165,   0.00000,   0.00000,   0,   0.00000,
1610  VectorVectorAdd_X86_I16_0512x0512x2,    3.40601,    2.89939,    0.84302,    7.14841,   0.00000,   0.00000,   0,   0.00000,
1611  VectorVectorAdd_X86_I16_0512x0512x2,    2.49707,    2.30046,    0.66382,    5.46135,   0.00000,   0.00000,   0,   0.00000,
1612  VectorVectorAdd_X86_I16_0512x0512x2,    2.85596,    6.44833,    0.85107,   10.15536,   0.00000,   0.00000,   0,   0.00000,
1613                    Statistics,         HD,         KE,         DH,         TT
1614                           Min,    1.88501,    2.27960,    0.58179,    5.08669,
1615                           Max,   19.33276,   14.54796,   15.84180,   29.11934,
1616                          Mean,    3.21496,    4.93552,    1.10981,    9.26029,
1617                        Median,    2.89709,    4.56527,    0.75952,    8.87686,
1618                Std.Deviation,    1.82650,    2.15783,    1.63380,    3.09075,
1619                      Std.Err,    0.06458,    0.07629,    0.05776,    0.10927,
1620

Normal text file          length : 4782704   lines : 25866    Ln : 1588   Col : 84   Sel : 0 | 0      UNIX      UTF-8 w/o BOM      INS
```

On executing the program, it generates a result.html file which displays the required details and graphs.

Angularjs and jquery are used for sorting the contents within a table, according to column.
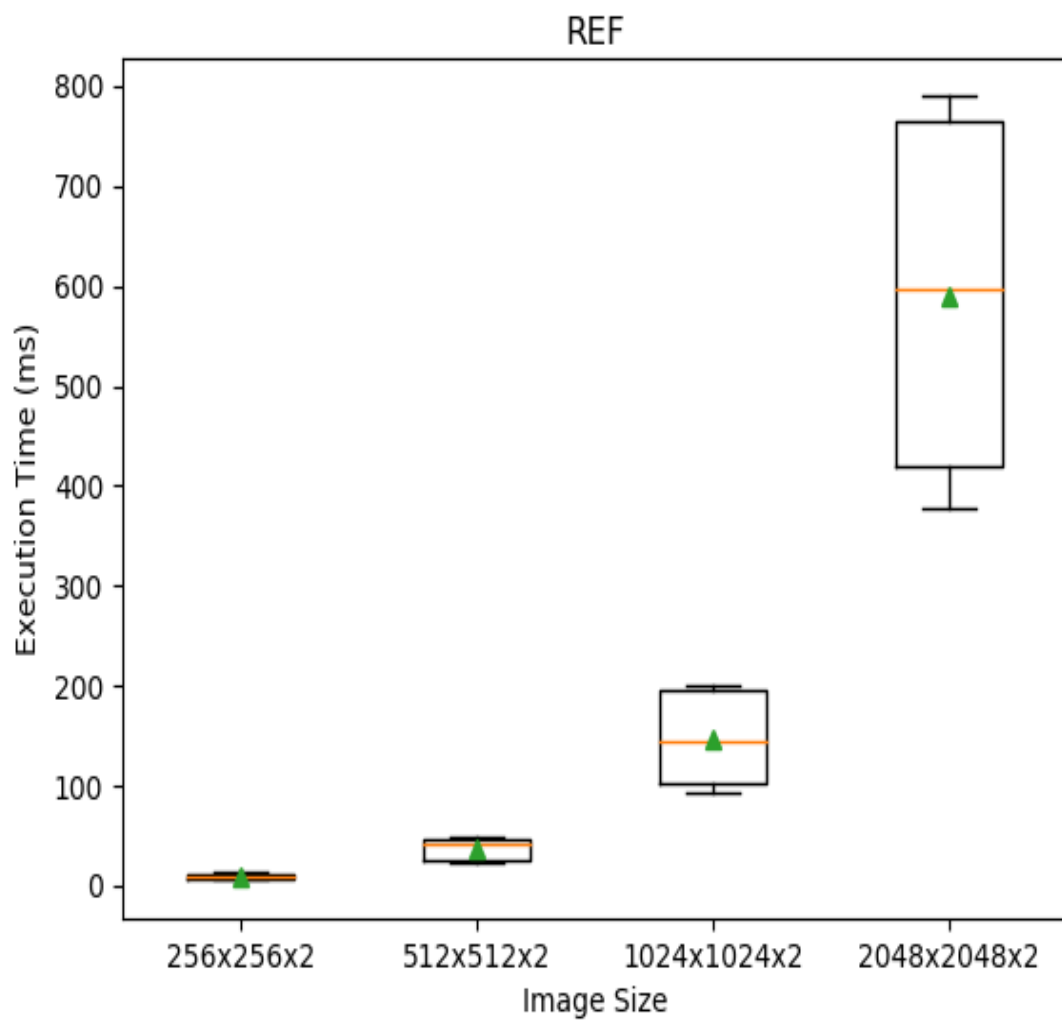
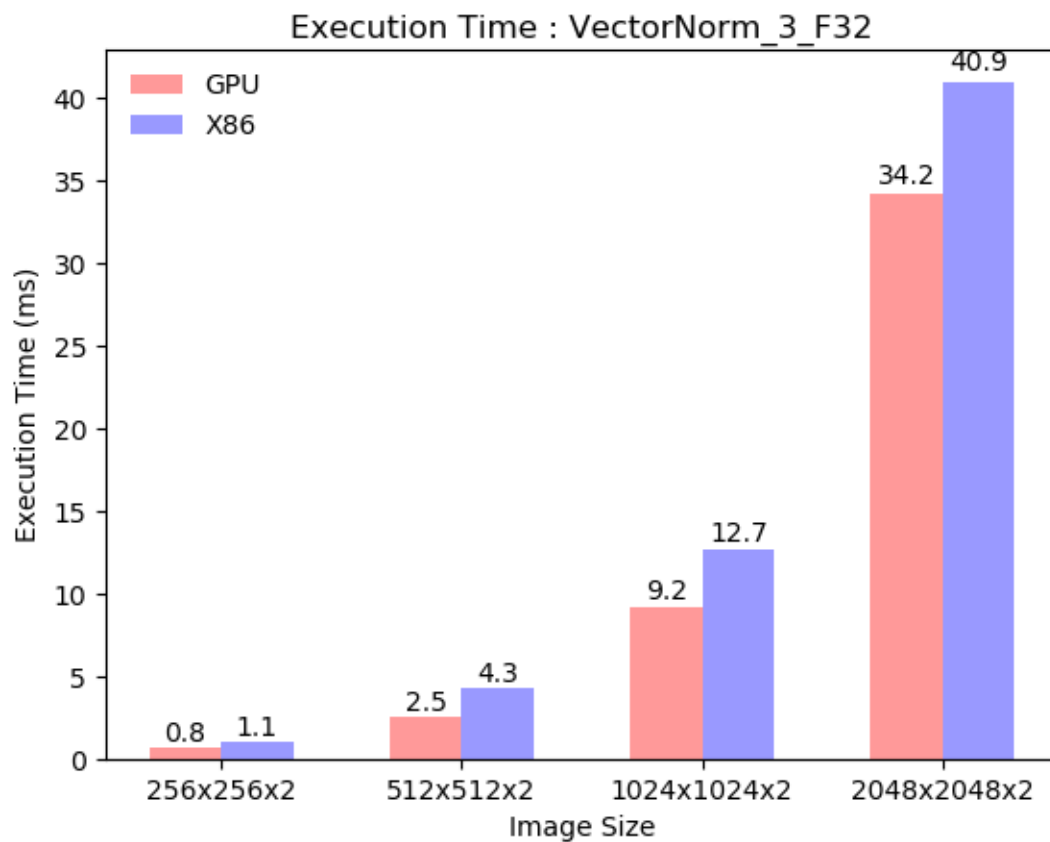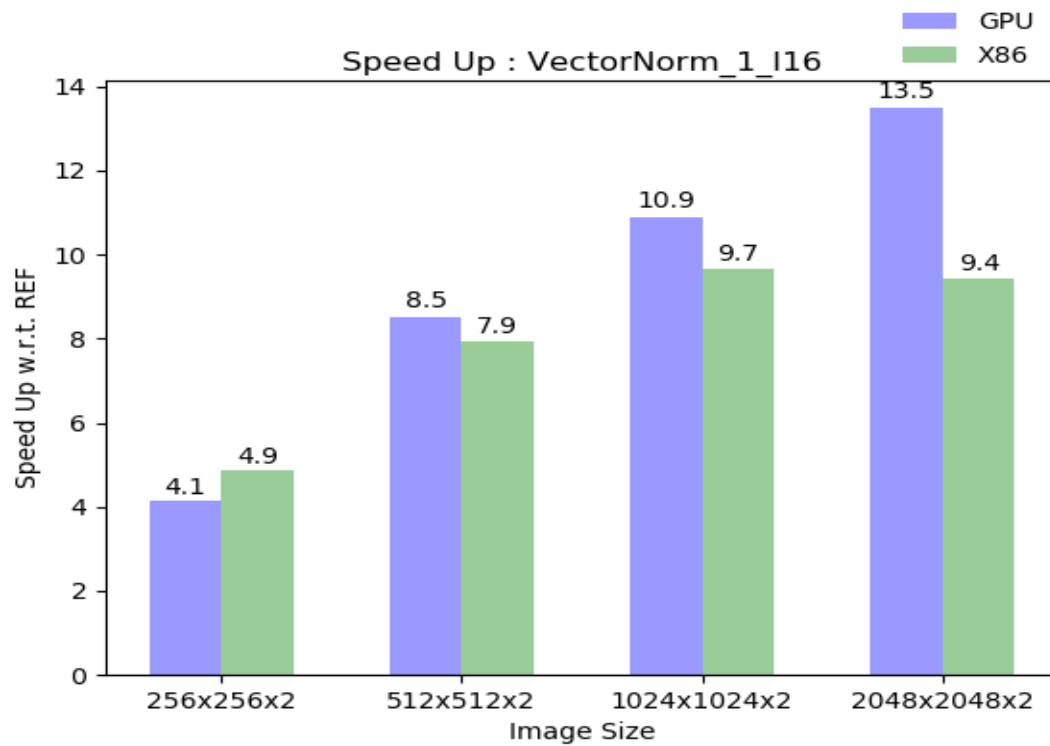Bootstrap, css are used for overall styling of the html file.

Python is the language in which whole code is written.

Matplotlib is used to generate comparison plots and box plots.

Code Output –

The graphs generated are displayed below -

REF

Execution Time (ms) vs Image Size

Speed Up : VectorNorm_1_I16

Execution Time : VectorNorm_3_F32

Results –

The program generated html file screenshots are displayed below –

Summary

Sl No	Test Group	Total Number of Test Cases	Passed	Failed	Success Ratio %
1	FunctionalTest	80	80	0	100.0
2	SupportedImages	80	80	0	100.0
3	NoisyImages	16	16	0	100.0
4	IllImages	16	16	0	100.0
5	RandomImageSize	16	16	0	100.0
6	SupportedImageSize	48	48	0	100.0
7	PerformanceTest	96	96	0	100.0
8	UnsupportedDataTypeImages	7	7	0	100.0
9	UnSupportedImgSize	6	6	0	100.0
10	UnSupportedAlgoParams	3	3	0	100.0
11	MismatchImageSize	6	6	0	100.0
12	Testing	55	55	0	100.0

TestGroup => FunctionalTest

Log

TestGroup => SupportedImages

Log

TestGroup => PerformanceTest

Log

Hide/show Images

Sl no	Scenario Name	Scenario Description	HD	KE	DH	TT	RMSE	BRMSE	No. Err	Max Err.	Mean Err.	Result Sort By :	SrcImage0	DstImage0	RefImage0
1	PerformTest1	+	0.00098	6.70605	0.00098	6.70801	0.00000	0.00000	0	0.00000	0.00000	passed	NA	NA	NA
2	PerformTest2	+	0.00098	27.22412	0	27.2251	0.00000	0.00000	0	0.00000	0.00000	passed	NA	NA	NA
3	PerformTest3	+	0.00098	103.71021	0.00098	103.71216	0.00000	0.00000	0	0.00000	0.00000	passed	NA	NA	NA
4	PerformTest4	+	0.00098	398.91382	0.00098	398.91577	0.00000	0.00000	0	0.00000	0.00000	passed	NA	NA	NA
5	PerformTest5	+	0.00122	5.74414	0	5.74536	0.00000	0.00000	0	0.00000	0.00000	passed	NA	NA	NA
6	PerformTest6	+	0.00098	23.3479	0.00098	23.34985	0.00000	0.00000	0	0.00000	0.00000	passed	NA	NA	NA
7	PerformTest7	+	0.00098	94.36621	0.00098	94.36816	0.00000	0.00000	0	0.00000	0.00000	passed	NA	NA	NA
8	PerformTest8	+	0.00098	366.86304	0.00098	366.86499	0.00000	0.00000	0	0.00000	0.00000	passed	NA	NA	NA
9	PerformTest9	+	0	13.03101	0.00098	13.03198	0.00000	0.00000	0	0.00000	0.00000	passed	NA	NA	NA
10	PerformTest10	+	0.00122	51.70581	0	51.70703	0.00000	0.00000	0	0.00000	0.00000	passed	NA	NA	NA
11	PerformTest11	+	0.00098	205.74805	0.00098	205.75	0.00000	0.00000	0	0.00000	0.00000	passed	NA	NA	NA
12	PerformTest12	+	0	812.16895	0	812.16895	0.00000	0.00000	0	0.00000	0.00000	passed	NA	NA	NA
13	PerformTest13	+	0.00098	13.04492	0	13.0459	0.00000	0.00000	0	0.00000	0.00000	passed	NA	NA	NA
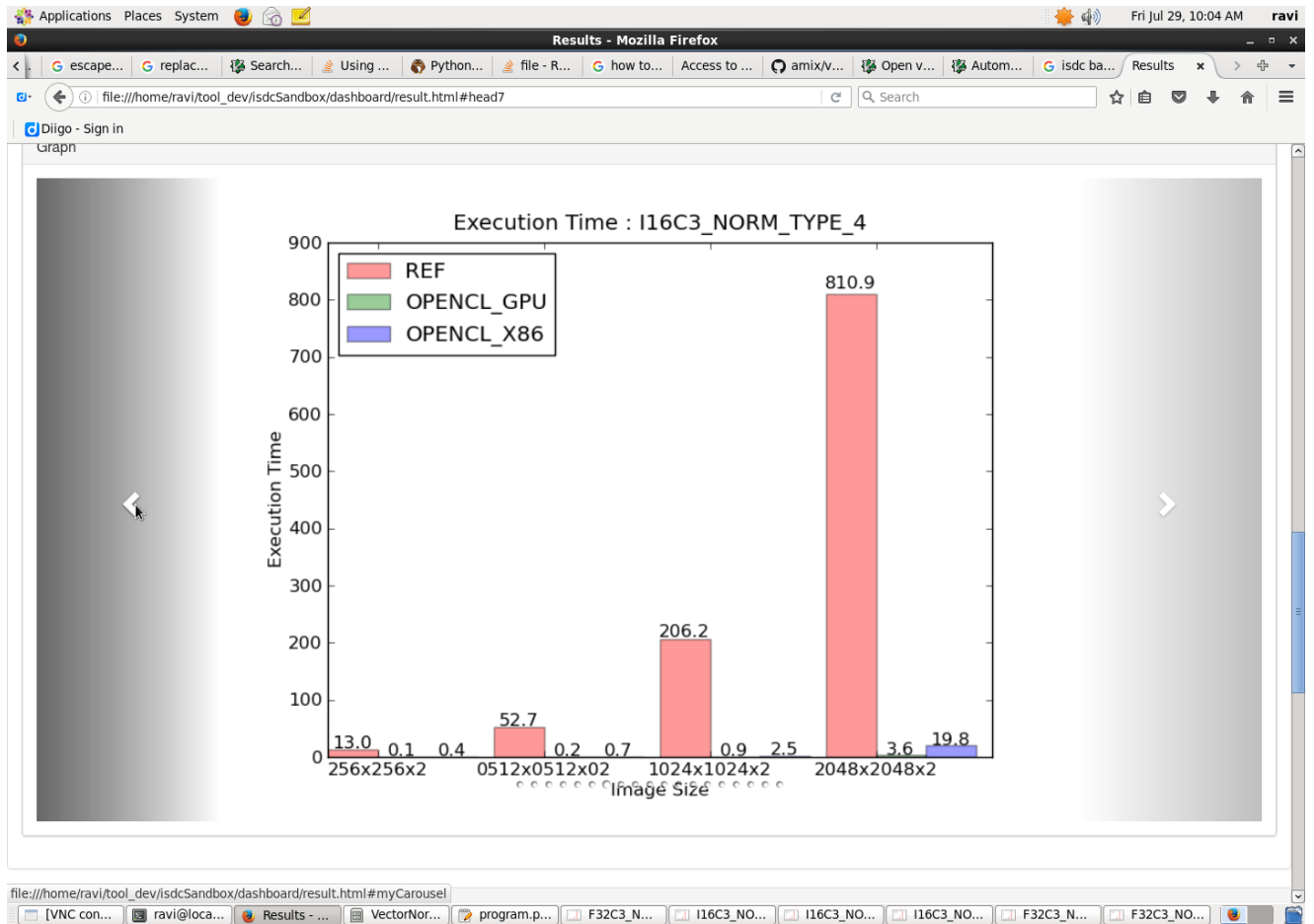
Tools Used

- Python 2.7
- Matplotlib
- Gedit, sublime
- Jquery, javascript
- Angularjs
- bootstrap, css, html

Conclusion –

- The readings and output were successfully obtained.
- Certain observations were made regarding these operators.
- The python programming code was able to generate the required files.