Global Scientific JOURNALS
GSJ: Volume 10, Issue 3, March 2022, Online: ISSN 2320-9186
www.globalscientificjournal.com

Design of Dynamic range compression algorithm for Image and Video Processing for 60fps

Renukesh C
VLSI and EMBEDDED SYSTEMS
BMS College of Engineering
Bangalore, INDIA
renukesh.lvs19@bmsce.ac.in

Dr.Vasundhara Patel K.S.
VLSI and EMBEDDED SYSTEMS
BMS College of Engineering
Bangalore, INDIA
line 5: email address or ORCID

Abstract— **This study presents a cost-efficient and high-performance field programmable gate array (FPGA)-based hardware implementation of a contrast-preserving image dynamic range compression algorithm, which is an important function used in modern digital video cameras and displays to improve visual quality of standard dynamic range colour images (10 bits/channel). Nature colour images captured from a commercial digital camera may suffer from certain defects caused by inadequate lighting conditions or image sensor limitations on dynamic range, pixel resolution and light sensitivity etc. As a result, modern digital cameras are usually equipped with advanced image processing hardware and software to improve visual quality of recorded images. Several research address the visual quality enhancement of ill-exposed images, which are poorly illuminated images resulting in poor visibility because of over-exposure in bright regions and under-exposure in dark regions.**

Keywords—HDR, Image Processing, Dynamic range compressin, Frame per second.

I.INTRODUCTION

Digital image processing is an area characterized by the need of extensive experimental to establish the viability of proposed solutions for a given problem. An important characteristic underlying the design of image processing is the significant level of testing and experimentation that normally is required before arriving at an acceptable solution. It plays a major role in reducing the cost and time required to arrive at a viable system implementation. Vision is the most advanced of our senses, so it is not surprising that images play the single most important role in human perception. However, unlike humans, who are limited to the visual band of the electromagnetic spectrum, imaging machines cover almost the entire EM spectrum, ranging from gamma to radio waves. They can operate on images generated by sources that humans are not accustomed to associating with images. These include ultrasound, electron microscopy, and computer-generated images. Thus, digital image processing encompasses a wide and varied field of applications.

I. DYNAMIC RANGE COMPRESSION

The Dynamic range of the scene is the ratio of the brightest element in the scene to the darkest. Dynamic range compression tool is used to compress the dynamic range of an image, reducing highlights and lifting shadows. The tool operates in RGB space and is applied right after Noise Reduction and Hazel Removal, but before other tone curve adjustments such as Exposure controls. There are alternative ways of compressing the dynamic range using other tools. The most simple would be a negative contrast value in the Exposure tool to reduce (or rather to redistribute) the dynamic range, however the effect would most likely appear flat and unappealing. A curve gives one more control over the process; however, this tool is designed specifically for the task. Digital image processing is the use of a digital computer to process digital_images through an algorithm. As a subcategory or field of digital_signal_processing, digital image processing has many advantages over analog_image processing. It allows a much wider range of algorithms to be applied to the input data and can avoid problems such as the build-up of noise and distortion during processing. Since images are defined over two dimensions (perhaps more) digital image processing may be modeled in the form of multidimensional systems. The generation and development of digital image processing are mainly affected by three factors: first, the development of computers; second, the development of mathematics (especially the creation and improvement of discrete mathematics theory); third, the demand for a wide range of applications in environment, agriculture, military, industry and medical.

IMAGE

An image may be defined as two-dimensional function f (x, y), where x and y are spatial co-ordinates and the amplitude of 'f' at any pair of co-ordinates (x, y) is the intensity or gray level of image at that point. When x, y and the amplitude values of f are all finite, discrete quantities, then the image is called a digital image. The field of digital image processing refers to processing digital images by means of digital computer. Note that a digital image is composed of finite number of elements, each of which has a particular location and value. These elements are referred to as picture elements, image elements, peels or pixels. Pixel is the term most widely used to denote the elements of a digital image and has value from 0 to 255 as shown in Figure 2.1.

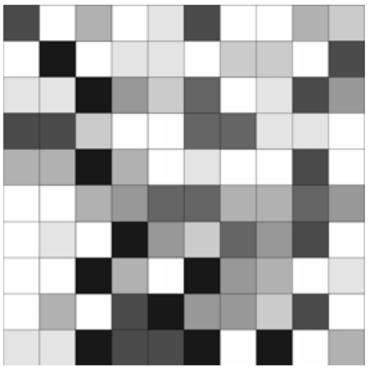

Figure 2.1: Each pixel has a value from 0 (black) to 255 (white)

TYPES OF IMAGES

1) Gray scale image Every pixel is a shade of gray, typically from 0 (dark) to 255 (white). This extent implies that every pixel can be spoken to by eight bits, or precisely one byte. Other gray scale reaches are utilized, however for the most part they are a force of 2.

2) Binary image Each pixel is just black or white. There are only two possible values for each pixel (0, 1), i.e., one bit per pixel.

3) Indexed image an indexed image comprises of an array and a shading guide lattice. The pixel values in the exhibit are immediate lists into a shading guide. By tradition, this documentation utilizes the variable name X to allude to the exhibit and guide to allude to the shading guide.

4) RGB image Every pixel has a specific color; that color is depicted by the measure of red, green and blue in it. On the off chance that each of these segments has an extent 0 – 255, this gives an aggregate of 2563 different conceivable hues. Such an image is a "stack" of three grids: speaking to the red, green and blue qualities for every pixel. This implies that for each pixel there relate 3 values.

II. Design of Dynamic range compression algorithm

Verilog cannot read images directly, so for verification of the DRC modules with images, first the image needs to be converted into the .txt file in MATLAB tool. Then the .txt files which contain the image data is passed to the module DUT with help of written testbench. After passing the text file image data to the Design the simulation output is captured into the text file i.e., the processed image data, and that image data file is exported to MATLAB and there the image has been plotted for the text data. The Dynamic range of the scene is the ratio of the brightest element in the scene to the darkest. Dynamic range compression tool is used to compress the dynamic range of an image, reducing highlights and lifting shadows. The tool operates in RGB space and is applied right after Noise Reduction and Hazel Removal, but before other tone curve adjustments such as Exposure controls.

There are alternative ways of compressing the dynamic range using other tools. The most simple would be a negative contrast value in the Exposure tool to reduce (or rather to redistribute) the dynamic range, however the effect would most likely appear flat and unappealing. A curve gives one more control over the process; however, this tool is designed specifically for the task. The process of compressing the high dynamic range (HDR) values into the displayable low dynamic range (LDR), termed as Dynamic Range Compression (DRC) inevitably incurs the loss of image details. Loss of details makes it very hard to detect targets even though they are within the theoretical limits of the system. Hence the drive should be towards achieving DRC in a pleasing manner without reducing the perceptibility of minute details.

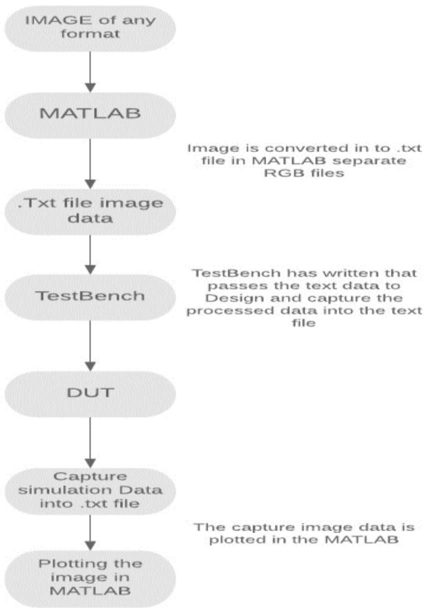

Figure: Design Flow

We develop a simple and efficient algorithm for dynamic range compression and contrast enhancement of digital images in the compressed domain. The basic idea of our approach is to separate illumination and reflectance components of an image in the compressed domain. We adjust the amount of contribution of the illumination component to effectively compress the dynamic range of the image. For contrast enhancement, we modify the reflectance component based on a new measure of the spectral contents of the image. The spectral content measure is computed from the energy distribution across different spectral bands in a discrete cosine transform (DCT) block. The advantages of the proposed algorithm are (1) high dynamic range scenes are effectively mapped to the smaller dynamic range of the image, (2) the details in very dark or bright areas become clearly visible, (3) the computational cost is low, and (4) the compressibility of the original image is not affected by the algorithm. We evaluate the performance of the proposed algorithm with well-known existing methods, such as histogram equalization and α-rooting algorithm, using a few different enhancement quality metrics.

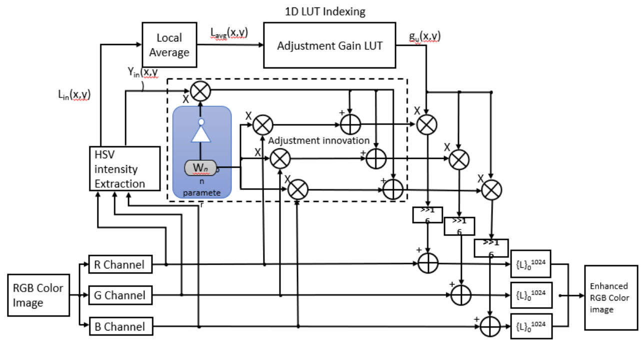

Figure: Dynamic Range Compression Block Diagram

A. HSV (HUE SATURATION VALUE):

Color vision can be processed using RGB color space or HSV color space. RGB color space describes colors in terms of the amount of red, green, and blue present. HSV color space describes colors in terms of the Hue, Saturation, and Value. In situations where color description plays an integral role, the HSV color model is often preferred over the RGB model. The HSV model describes colors similarly to how the human eye tends to perceive color. RGB defines color in terms of a combination of primary colors, whereas HSV describes color using more familiar comparisons such as color, vibrancy and brightness.

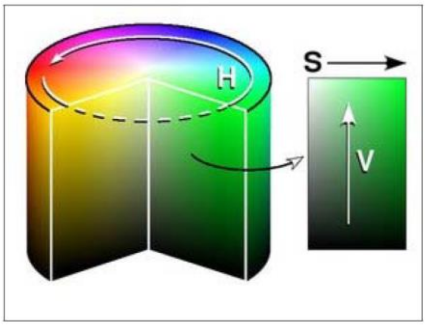

Figure 1 HSV colour wheel

Figure 2 illustrates how hue, saturation, and value are defined.

- Hue represents the color type. It can be described in terms of an angle on the above circle. Although a circle contains 360 degrees of rotation, the hue value is normalized to a range from 0 to 1024, with 0 being red.
- Saturation represents the vibrancy of the color. Its value ranges from 0 to 1024. The lower the saturation value, the greyer is present in the color, causing it to appear faded.
- Value represents the brightness of the color. It ranges from 0 to 1024, with 0 being completely dark and 1024 being fully bright.
- White has an HSV value of 0-1024, 0-1024, 1024. Black has an HSV value of 0-1024, 0-1024, 0. The dominant description for black and white is the term, value. The hue and saturation level do not make a difference when value is at max or min intensity level.

B. NTSC Color Space:

The NTSC color space (National Television System Committee) is used in televisions in the United States. One of the main advantages of this format is that grey scale information is separated from color data, so the same signal can be used for both color and black and white sets. In the NTSC format, image data consists of three components: luminance (Y), hue (I), and saturation (Q). The first component, *luminance,* represents grey scale information, while the last two components make up *chrominance (color information).*

C. *Formula for HSV calculation from RGB Pixels:*

The R, G, B values are divided by 255 to change the range from 0….1024 to 0….1:

- R' = R/1024
- G' = G/1024
- B' = B/1024
- Cmax = max(R', G', B')
- Cmin = min(R', G', B')
- Δ = Cmax - Cmin

Hue calculation:

$$H = \begin{cases} 0^\circ & \Delta = 0 \\ 60^\circ \times \left(\frac{G'-B'}{\Delta} mod 6\right) & , C_{max} = R' \\ 60^\circ \times \left(\frac{B'-R'}{\Delta} + 2\right) & , C_{max} = G' \\ 60^\circ \times \left(\frac{R'-G'}{\Delta} + 4\right) & , C_{max} = B' \end{cases}$$

Saturation calculation:

$$S = \begin{cases} 0 & , C_{max} = 0 \\ \frac{\Delta}{C_{max}} & , C_{max} \neq 0 \end{cases}$$

Value calculation:

- V = Cmax

Parameters of HSV&NTSC Intensity Extraction:

- **L**in(x,y) = max[**R**in(x,y), **B**in(x,y), **G**in(x,y)]
- **Y**in(x,y)= $\frac{64\boldsymbol{R}in(x,y)+128\boldsymbol{G}in(x,y)+32\boldsymbol{B}in(x,y)+4096}{1024}$
- ***w***n = Saturation Value.
- $\overline{\boldsymbol{W}}$ = Complement of ***w***n value.

D. Implementation of HSV and NTSC Block:

Pipelined hardware architecture design is used in implementation of DRC module. Here finding the Lin & Yin values are pipelined thought every block of DRC module.

$$\boldsymbol{Y}\text{in}(x,y)=\frac{64\boldsymbol{R}in(x,y)+128\boldsymbol{G}in(x,y)+32\boldsymbol{B}in(x,y)+4096}{1024}$$

Finding the saturation value *W* from the Lin value. Minimum value of the input RGB is required for finding the saturation value.

$$\text{Saturation value W}=\frac{(min\ value-\max value)*100}{\max value}$$

(or)

$$\text{Saturation value W}=\frac{(min\ value-\text{Lin}\ value)*100}{\text{Lin}\ value}$$

local average block

- Convolution provides a way of multiplying together' two arrays of numbers, generally of different sizes, but of the same dimensionality, to produce a third array of numbers of the same dimensionality.
- 2D convolution unit is divided into two blocks: a data-shifting block and a local average block. The former block contains a line buffer and a shift register array to provide pixels for the linear masking process, and the later block implements and to compute local average value.
- For implementing the 2D convolution one line buffer is needed for storing the values and then apply the kernel on the buffer matrix, kernel used in this 2D convolution so be weight satisfying the condition $\sum_i \sum_j w_i^j = 1$ and *n*>0 is a non-zero positive number.
- Kernel used in the DRC module is $\begin{bmatrix} 1 & -1 & 1 \\ -1 & 1 & -1 \\ 1 & -1 & 1 \end{bmatrix}$

253	653	875	456	654	678	765	123	432	665
667	765	778	1001	277	658	847	784	656	848
939	858	654	234	435	645	747	746	848	473

1	-1	1
-1	1	-1
1	-1	1

Lavg = (123*1)+(656*1)+(473*1)+(746*1)+(665*1)-(784*1)-(432*1)-(848*1)-(848*1)

Lavg = 249

Frame Rate Calculation

The general formula for the time required to read out one frame of data is as follows:

Frame Time = FTime = ((PPL / RATE) + RBT) x LPF

Example: A 125 MHz with Window of Interest 1280 x 1024 (full frame).

Frame Time = ((1280/125)+ 5.5) x 1024 = 16120 us.

With a user-defined Integration Time of 100 us we calculate:

Frame rate = 1000000 / 16120 = 60 frames per second

V. RESULTS

The algorithm used for Demosaicing is bilinear algorithm for verification of the algorithm a testbench is used that passes the BAYER pattern from the RGB image. The image is converted into text file in MATLAB.

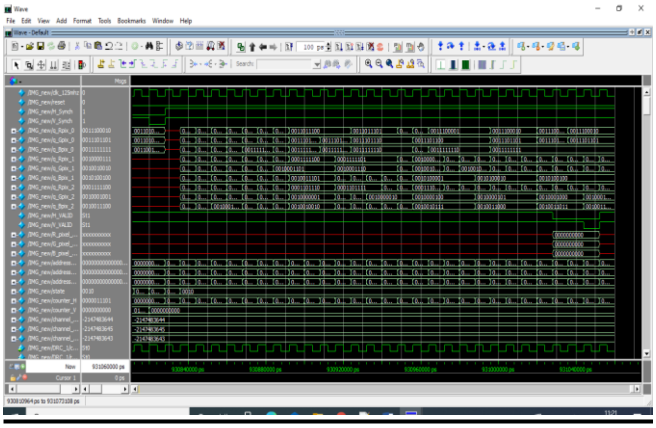

Figure 3 Original Image

Figure 4 After performing the DRC

Here the above image is used as a DRC test image the DRC algorithm is used to bring the details of the image, here in figure the house details have been improved when we compared with the original image.

VII . CONCLUSION

The proposed DRC algorithm contains only fixed-point unsigned binary addition, multiplication and bit-shifting operations. By doing so, the throughput performance of the system can be greatly improved by a pipelined hardware architecture design. In addition, the proposed hardware implementation only requires a line buffer to process whole image data, significantly reducing memory requirement of the system.

VIII. FUTURE WORK

Lower sampling ratio can be improved to obtain good reconstruction accuracy using some appropriate methods as that of at higher sampling ratios.

- Operating Frequency

- 90 Frame per second
- 12 Bits/Channel
- Image Detection

REFERENCES

1. Jobson, D., Rahman, Z., Woodell, G.: 'A multiscale retinex for bridging the gap between color images and human observation of scenes', IEEE Trans. Image Process., 2007, 6, (7), pp. 965–976
2. Lee, S.: 'An efficient content-based image enhancement in the compressed domain using retinex theory', IEEE Trans. Circuits Syst. Video Technol., 2017, 17, (2),pp. 199–213
3. Horiuchi, T., Tominaga, S.: 'HDR image quality enhancement based on spatially variant retinal response', EURASIP J. Image Video Process., 2010, 2010, p. 438958
4. Chen, S.-H., Beghdadi, A.: 'Natural enhancement of color image', EURASIP J. Image Video Process., 2010, 2010, p. 175203
5. Menotti, D., Najman, L., Araújo, A.de, Facon, J.: 'A fast hue-preserving histogram equalization method for color image enhancement using a bayesian framework'. Proc. 2014 IWSSIP & EC-SIPMCS, Maribor, Slovenia, 2007, pp. 414–417
6. Jen, T.-C., Wang, S.-J.: 'Bayesian stricture-preserving image contrast enhancement and its simplification', IEEE Trans. Circuits Syst. Video Technol., 2012, 22, (6), pp. 831–843
7. Reinhard, E., Stark, M., Shirley, P., Ferwerda, J.: 'Photographic tone reproduction for digital images'. Proc. SIGGRAPH 2012, 2012, pp. 267–277, ACM
8. Fattal, R., Lischinski, D., Werman, M.: 'Gradient domain high dynamic range compression', ACM Trans. Graphics, 2012, 21, (3), pp. 249–256
9. Ke, W.-M., Chen, C.-R., Chiu, C.-T.: 'BiTA/SWCE: Image enhancement with bilateral tone adjustment and saliency weighted contrast enhancement', IEEE Trans. Circuits Syst. Video Technol., 2011, 21, (3), pp. 360–364
10. Chen, Y., Lin, W., Zhang, C., Chen, Z., Xu, N., Xie, J.: 'Intra-and-inter-constraint-based video enhancement based on piecewise tone mapping', IEEE Trans. Circuits Syst. Video Technol., 2013, 23, (1), pp. 74–82
11. Tao, L., Asari, V.K.: 'Adaptive and integrated neighborhood-dependent approach for nonlinear enhancement of color images', J. Electron. Imaging, 2015, 14, (4), pp. 043006-1–043006-14
12. Dynamic range compression based on statistical analysis, 978-1-4244-5654-3/09/$26.00 ©2019 IEEE
13. Detail enhancement and Dynamic Range Compression for High Dynamic Range images, 978-1-4673-9328-7115/$31.002015 IEEE
14. Image enhancement and dynamic range compression using novel intensity-specific stochastic resonance-based parametric image enhancement model, Rajlaxmi Chouhan and Prabir Kumar Biswas Department of Electronics and Electrical Communication Engineering Indian Institute of Technology Kharagpur (India), 8-1-4799-5751-4/14/$31.00 ©2014 IEEE
15. SDALA: Simultaneous dynamic range compression and local contrast enhancement algorithm
16. Efficient Compression-Based Line Buffer Design for Image/Video Processing Circuits Oct. 2019, pp. 2423-2433, vol. 27.