ISSN: 0975-3583, 0976-2833 VOL 09, ISSUE 01, 2018

# DESIGN AND IMPLEMENTATION OF HISTOGRAM ESTIMATION ARCHITECTURES FOR 8X8 IMAGE

1. Dr.K.Gouthami
Professor and Dean of ECE Department
Malineni Lakshmaiah Women's Engineering College,
Pulladi gunta, Guntur,India- 522017
e-mail:malineni.ece@gmail.com

2. Dr.P.Nageswar Rao
Director and Professor of ECE Department
Malineni Lakshmaiah Women's Engineering College,
Pulladi gunta, Guntur,India- 522017
e-mail:pnrao33@gmail.com

## Abstract

Histogram estimation designs for evaluating sampled data have recently gained popularity and have found several applications in image processing, communications, computer vision, and pattern recognition, among other fields. As part of this study, the suggested histogram estimation architecture is utilised to count the bin values. Additionally, the architecture is implemented on an Application Specific Integrated Circuit (ASIC) platform and also on an FPGA platform to demonstrate its functionality (FPGA). Through the use of MATLAB, the input 8x8 picture is translated into binary samples, and these values are then passed on to the input of the verilog code. It is the modelsim programme that is used to generate the simulation results for the suggested design. For the 8x8 image, we evaluate the performances of both existing and proposed architectures, and to obtain the performance of ASIC, the architectures are implemented in cadence encounter using 180nm technology. The percentage reduction of area power and delay is 17.6 percent, 32 percent, and 18 percent, respectively for the existing and proposed architectures. In addition, the virtex-6 device reduces the consumption of lookup tables, slice registers, and the amount of flipflops per unit of time.

Optimal Bin Counter, Application Specific Integrated Chip, Field Programmable Gate Array, Dual port Read Only Memory, Histogram estimate are some of the terms used in this paper.

## 1. Introduction

It is becoming more relevant in different applications such as image processing, medical imaging, pattern recognition, and machine learning to evaluate probability density functions (PDFs) based on sampling statistics [1, 2]. When constructing an estimate of the density function from the observed data samples [2, one of the essential ideas in statistics is PDF, which is one of the most fundamental notions in statistics. In general, PDF estimate approaches may be divided into two types: nonparametric methods and parametric methods [3, which are discussed below]. There are many different kinds of density

ISSN: 0975-3583, 0976-2833 VOL 09, ISSUE 01, 2018

functional forms that may be studied using nonparametric techniques. When the PDF nature is unknown or shifting parametric models are used, the precision of the results might be unfruitful [4].

Adobe PDFs have a wide variety of applications in a wide range of fields, including image processing, machine learning, and communications. Both histogram estimation and kernel estimation function [5] are nonparametric approaches that may be divided into two types. A stepwise response is provided by the former estimator, which produces continually quantized approximates of the real PDF [6].

A histogram is a graphical depiction of the distribution of data that has been sampled. In image processing, the histogram offers a number of benefits, notably in that it may be used to improve the contrast of an image and to choose the appropriate threshold value for thresholding an image [7]. In this research, an architecture for nonparametric PDF assessment is provided, which is based on the histogram-based technique. Although histogram estimation techniques are often displayed using simulation tools such as MATALB, it is not viable to implement the architecture for histogram creation [8]. The architecture makes use of the many resources available on today's FPGAs while maintaining a highly parallelized and pipelined design environment. When working with sampled data, it is capable of doing real-time assessment at greater rates while extracting a range of statistical features [9].

The following is how the work will be organised. Section 2 discusses related work on conventional histogram architectures based on FPGAs, which includes FPGA-based conventional histogram designs. Section 3 provides an introduction and explanation of the intended work. Section 4 demonstrates how to conduct a performance study of designs. Section 5 concludes with a discussion of the future scope of the project.

## 2. Related works

In this research [10], a histogram estimation architecture was built to compute the histogram for any grayscale picture with the use of MATLAB, and the architecture was then implemented in both FPGA and PSoC hardware, as well as in a variety of other platforms. The architecture is composed of two primary modules: the first is responsible for the creation of the histogram, and the second is responsible for displaying the histogram bin structure on the screen. It is possible to increase the hardware resource consumption of histogram estimation of grayscale pictures by using two independent modules in an FPGA implementation, for example.

ISSN: 0975-3583, 0976-2833 VOL 09, ISSUE 01, 2018

Using FPGAs to implement computer vision application algorithms, we investigated the architectures utilised to create histogram estimates for histogram estimations in this study [11]. A number of simplification designs for the oriented gradient algorithm were discussed in the study, which aimed to improve overall performance by reducing design complexity and increasing design speed. Last but not least, the author offered three design principles for FPGA-based oriented gradient implementation for a wide range of applications. Nonetheless, while the hardware implementation of FPGAs, notably for the virtex-6 device, is more complex, the accuracy of histogram estimate is high.

Using this work [12], the author demonstrated an architecture for speeding up the process of histogram estimation. In particular, the ability to detect quick facial identification even when the picture changes suddenly owing to ambient temperature is a significant benefit of this approach. As a result of the implementation on the Zynq Z-7020 FPGA, the designs need a somewhat larger amount of space than the previous versions.

## 3. Proposed Histogram Architecture

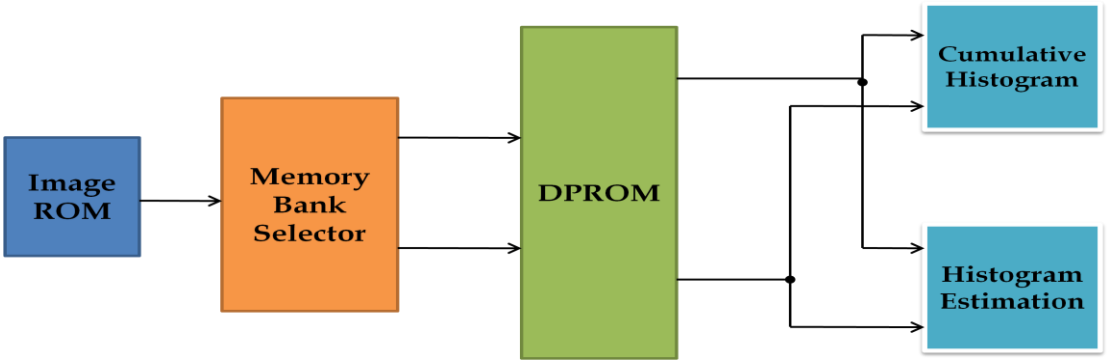


**Figure 1. Proposed Histogram Architecture Process**

FIGURE 1 depicts the proposed histogram architecture, which is comprised of the memory bank selection, Dual Port ROM, memory bank selector, and counters. The procedure of estimating the histogram of 64 samples is detailed in detail in this section. To begin, a MATLAB script is used to read an input picture of size 8x8 that comprises 64 samples in order to assess the histogram estimation procedure. Through the use of the "dec2bin" function, the pixel values of the picture are converted into binary data.. For

ISSN: 0975-3583, 0976-2833 VOL 09, ISSUE 01, 2018

evaluation of the histogram estimation process on the hardware platform, the converted binary data comprises 8 bits of data saved in a text file sent to the hardware platform.

The 16 pixel sample values are depicted in figure 2.

| | | | |
|---|---|---|---|
| 00000001 | 00000100 | 00000000 | 00000001 |
| 00000111 | 00000110 | 00000110 | 00001001 |
| 00000101 | 00000111 | 00000011 | 00000101 |
| 00000101 | 00000111 | 00000011 | 00001111 |

**Figure 2. Input values of image**

The FPGA platform is used to perform the histogram estimation method for 64 samples. The counter in this block diagram is created in such a manner that it counts pixel values ranging from 0 to 63, as shown in the figure.

The samples are sent to the dual port ROM, which runs the pattern process based on the least significant values determined by the counter. Table 1 contains the values for the pattern values.

**Table 1. Pattern values**

| Address | Pattern |
|---|---|
| 0 | 111111….....1111111 |
| 1 | 011111……..1111111 |
| 2 | 001111….....1111111 |
| 3 | 000111……..1111111 |
| 4 | 000011……..1111111 |
| . | . |
| . | . |
| . | . |
| 254 | 000000……..0000011 |
| 255 | 000000…….0000001 |

Initially, dual port ROM executes the pattern values of 16sample[0] and 16sample[1], where the memory values of these samples are 1 and 7 respectively. The output values of dual port ROM are given in equation 2 and equation 3.

$$16sample[0]\_dualportROM\_out = 11111.....111111$$

ISSN: 0975-3583, 0976-2833 VOL 09, ISSUE 01, 2018

(2)

$$16sample[1]\_dualportROM\_out = 01111.....111111 \quad (3)$$

According to the clock signal pattern process is executed for all the samples. The output of dual port ROM is fed to the optimal bin counter which is used to count the number of one's appear in the 256 bit values.

## 4. Results and Discussion

This section discusses the performance analysis of existing and planned histogram estimation designs, as well as their advantages and disadvantages. Figures 3 and 4 depict the input picture and the sample values that correspond to it, respectively, in this example. The two designs are designed for an 8x8 picture size with 64 samples, which is the size of the two architectures. Modelsim 10.3 is used to simulate existing and new histogram structures in order to examine the result of the histogram estimation algorithm. As shown in Table 2, the performance of FPGA is obtained from cadence encounter using 180nm technology, while the performance of ASIC is obtained from cadence encounter using the same technology. The architectures are also implemented in Xilinx ISE 14.5 to obtain the performance of FPGA such as look up tables, slice registers, lookup table flip-flop pairs, and frequency, while the performance of ASIC is obtained from cadence encounter using 180nm technology.

**Figure 3. 8x8 input image**

00000001
00000111
00000101
00000101
00000100
00000110

ISSN: 0975-3583, 0976-2833 VOL 09, ISSUE 01, 2018

00000111
00000111
00000000
00000110
00000011
00000011
00000001
00001001
00000101
00001111

**Figure 4. sample values**

**Table 2. Performance analysis of ASIC using 180 nm technology**

| Architecture | Area ($um^2$) | Power (W) | Delay (ps) | APP ($um^2$×W) | ADP ($um^2$×ps) |
|---|---|---|---|---|---|
| **Existed Architecture_64 Samples** | 1472301.5 | 6.24 | 242 | 9187161.36 | 356296963 |
| **Proposed Architecture_64 Samples** | 1212868 | 4.87 | 198 | 5906667.16 | 240147864 |

**Table 3. Percentage of ASIC performance reduction by Proposed Architecture**

| Parameters | Percentage of reduction by Proposed Architecture |
|---|---|
| **Area** | 17.6% |
| **Power** | 32% |
| **Delay** | 18% |

Table 3 shows the amount of percentage reduction achieved by the proposed histogram estimation architecture, and Table 4 shows the performance analysis of the Xilinx Virtex-6 FPGA device. Table 3: Amount of percentage reduction achieved by suggested histogram estimation architecture

**Table 4. Performance analysis of FPGA using Virtex 6 device**

| FPGA performances | Total resources | Occupied resources | | % of utilization | |
|---|---|---|---|---|---|
| | | Existed | Proposed | BHE | MHE |
| **slice registers** | 93120 | 1062 | 1062 | 1% | 1% |

ISSN: 0975-3583, 0976-2833 VOL 09, ISSUE 01, 2018

| | | | | | |
|---|---|---|---|---|---|
| **slice LUTs** | 46560 | 11974 | 1672 | 25% | 3% |
| **fully used LUT-FF pairs** | 12500 | 536 | 601 | 4% | 4% |
| **Frequency** | 156 | 8 | 8 | 5% | 5% |

. The operation speed is high and delay is less for the MHE architecture due to the higher operating frequency i.e., 112.335MHz.

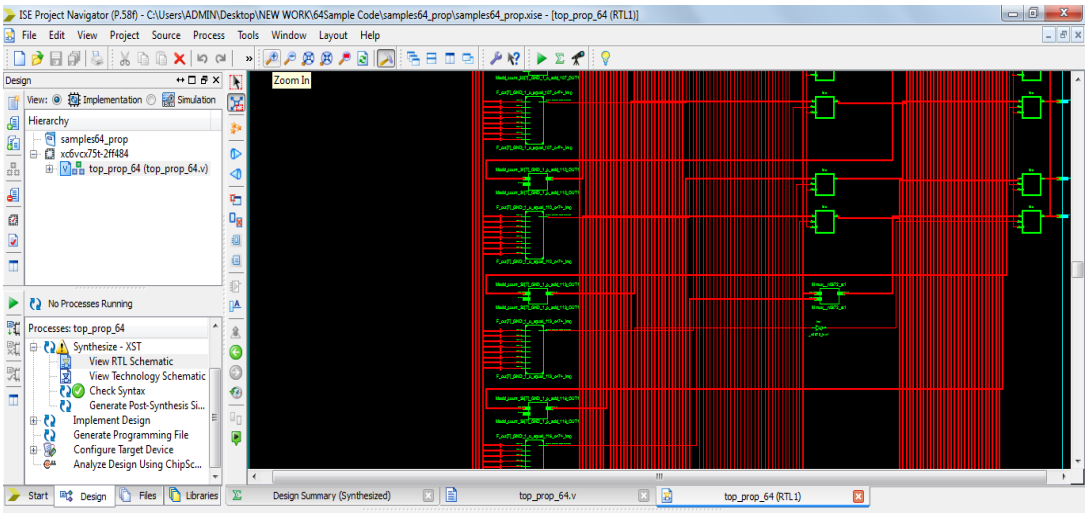


**Figure 5. RTL Schematic of Histogram estimation**

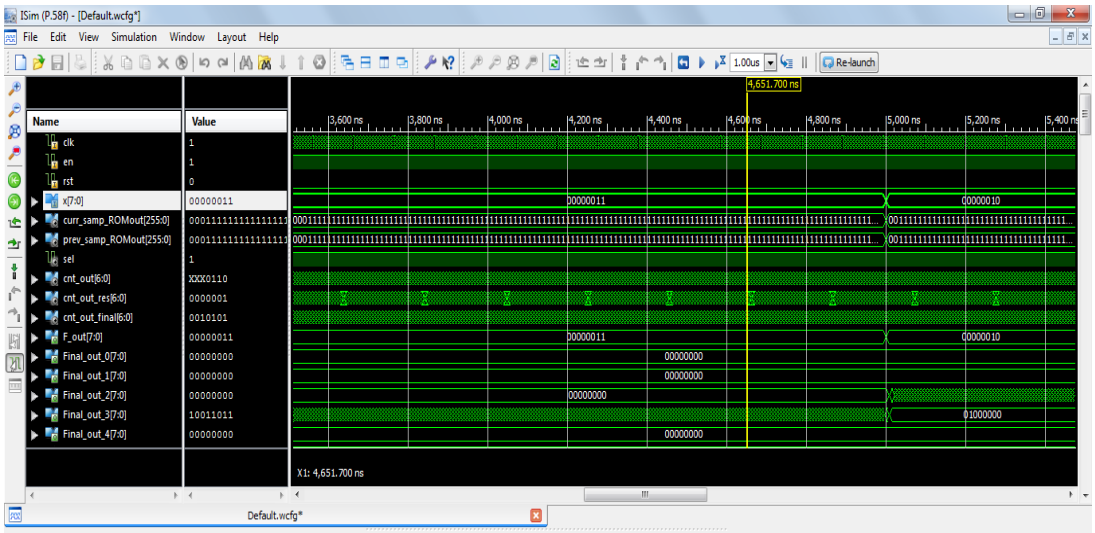


**Figure 6. Histogram estimation simulation output**

ISSN: 0975-3583, 0976-2833 VOL 09, ISSUE 01, 2018

Figure 5 and 6 shows the RTL schematic and Simulation results of proposed histogram estimation structure respectively.

## 5. Conclusion

It is shown in this paper that the proposed histogram architecture can be utilised to accurately simulate the histogram of an 8x8 grayscale picture. It is possible to increase the performance of histogram estimation by using a dual port ROM as well as a carry select adder. When compared to the existing histogram design, the suggested architecture only requires a single dual port ROM to hold both the old and new samples, while the existing histogram architecture requires two ROM blocks. When compared to the existing design, the suggested architecture has a frequency of 112.335MHz, which implies that the operating speed of the architecture has been enhanced and the latency has been minimised. When compared to the existing design, the performance of FPGA is significantly enhanced in terms of LUTs, slice registers, and LUT flip flop pairs. The suggested histogram estimation architecture reduces the quantity of area, latency, and power by a percentage of 17.6 percent, 32 percent, and 18 percent, respectively, when 64 samples are used.

## References

[1] Yin, S., Ouyang, P., Chen, T., Liu, L. and Wei, S., 2015. A configurable parallel hardware architecture for efficient integral histogram image computing. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 24(4), pp.1305-1318.

[2] Pokrajac, Dragoljub, et al. "Variable Width Elliptic Gaussian Kernels for Probability Density Estimation." Telecommunications in Modern Satellite, Cable and Broadcasting Services, 2007. TELSIKS 2007. 8th International Conference on. IEEE, 2007.

[3] Carbone, P., and D. Petri. "Non-parametric estimation of probability density functions via a simple interpolation filter." Instrumentation and Measurement Technology Conference (I2MTC), 2015 IEEE International. IEEE, 2015.

[4] Hai-Yan, Luan, and Jiang Hua. "Vicinal risk minimization based probability density function estimation algorithm using SVM." Information and Computing (ICIC), 2010 Third International Conference on. Vol. 4. IEEE, 2010.

[5] Pană, Cristian, Stefano Severi, and Giuseppe ThadeuFreitas de Abreu. "An adaptive approach to non-parametric estimation of dynamic probability density functions." Positioning, Navigation and Communications (WPNC), 2016 13th Workshop on. IEEE, 2016.

ISSN: 0975-3583, 0976-2833 VOL 09, ISSUE 01, 2018

[6] Hanebeck, Uwe D. "Kernel-based deterministic blue-noise sampling of arbitrary probability density functions." Information Sciences and Systems (CISS), 2014 48th Annual Conference on. IEEE, 2014.

[7] Praveen V, Sahitya G, Krishnasree and C.D. Naidu. FPGA implementation of image enhancement algorithms for biomedical image processing, International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, 2(11), 2013, pp. 5749-5753.

[8] Tarek M. Bittibssi, Gowda I Salma, Yehia Z Mehaseb and Adel E. Henawy. Image enhancement algorithm using FPGA. International Journal of Computer Science and Communication Networks, 2(4), pp. 536-542.

[9] Yang, Y., Liu, Y.X. and Dong, Q.F., 2017. Sliced integral histogram: an efficient histogram computing algorithm and its FPGA implementation. Multimedia Tools and Applications, 76(12), pp.14327-14344.

[10] Hazra, S., Ghosh, S., Maity, S.P. and Rahaman, H., 2016. A new FPGA and programmable soc based VLSI architecture for histogram generation of grayscale images for image processing applications. Procedia Computer Science, 93, pp.139-145.

[11] Ghaffari, S., Soleimani, P., Li, K.F. and Capson, D.W., 2020. Analysis and Comparison of FPGA-Based Histogram of Oriented Gradients Implementations. IEEE Access, 8, pp.79920-79934.

[12] Bonny, T., Rabie, T. and Hafez, A.A., 2018. Multiple histogram-based face recognition with high speed FPGA implementation. Multimedia Tools and Applications, 77(18), pp.24269-24288.