

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 7, Issue 5, May 2018

Stochastic Implementation of Sauvola Local Image Thresholding Algorithm

Athira Thambi¹, Merlin Thomas²

M.Tech Scholar, Dept. of ECE, Viswajyothi College of Engineering and Technology, Vazhakulam, Kerala, India¹

Asst. Professor, Dept. of ECE, Viswajyothi College of Engineering and Technology, Vazhakulam, Kerala, India²

ABSTRACT: Most document analysis algorithms are built on taking advantage of underlying binarized image data. Among the various binarization techniques available, Sauvola local image thresholding algorithm shows excellent binarization performance for degraded images. However this algorithm is computational intensive and subjected to internal circuit noises. A low cost alternative to conventional binary computing is stochastic computing. Stochastic computing uses streams of random bits to perform computation using conventional digital logic gates and can guarantee reliable computation using unreliable devices. Our experimental results show that, the stochastic implementation of Sauvola algorithm is more tolerant to faults and consumes less time.

KEYWORDS: Stochastic computing, binarization, sauvola local image thresholding

I. INTRODUCTION

Image binarization is one of the principal problems of the image processing applications. For extracting useful information from an image we need to divide it into distinctive components e.g. background and foreground objects for further analyses. Often the gray level pixels of the foreground objects are quite different from background. The simplest way to accomplish this binarization is thresholding.

Thresholding is one of the basic image processing operations. It is the simplest method of segmentation which allows extraction of meaningful objects from the background. A standard thresholding procedure uses a single parameter called a threshold value. Thresholding can be of two types, global image thresholding and local image thresholding. Among the most commonly used ones, the global thresholding method, Otsu's method (Otsu, 1979), computes the threshold based on statistical parameters of the image's histogram. Global thresholding which uses a single threshold for the whole image proves to be very good for simple images with relatively uniform illumination. But it can be problematic for images with non-uniform illumination.

For document images of a good quality, global thresholding is capable of extracting the document text efficiently. For document images suffering from different types of document degradation we estimate a local threshold for each document image pixel which is usually capable of producing much better binarization results [2]. Because of the good results on document images and is widely used in practice, this paper focuses on Sauvola's binarization. However this algorithm is computational intensive and subjected to internal circuit noises. A low cost alternative to conventional binary computing is stochastic computing. Stochastic computing can reduce the computation time to a greater extent and also it reduces the number of errors at low cost compared to conventional Sauvola implementation.

In this paper we attempt a study on the stochastic implementation of Sauvola local image thresholding algorithm. The rest of the paper is structured as follows. In Section II and III we give an overview of local image thresholding and stochastic computing and discuss on the proposed method. In Section IV we discuss about the stochastic implementation. Section V shows some results of the proposed method applied to real documents and compare them to other methods' results. We conclude on the achievements of this work in Section V.

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 7, Issue 5, May 2018

II. LITERATURE SURVEY

The most straightforward strategy for image analysis uses the brightness of regions in the image as a means of identification. It is assumed that the same type of feature will have the same brightness throughout the whole image. Uniform illumination, which allows easier and more reliable image segmentation, can be achieved mainly indoors by using artificial light sources and isolating the subject from sunlight, light reflexes, etc.

The paper introduces a local thresholding algorithm implementation on an FPGA device. Thresholding is one of the basic image processing operations. It is the simplest method of segmentation, allowing extraction of meaningful objects from the background. A standard thresholding procedure uses a single parameter called a threshold value. Global thresholding which establish a single threshold for the whole image proves to be very good for simple images with relatively uniform illumination. Images with non-uniform illumination can be problematic. So as to achieve reliable thresholding results, a more complex method is required, either local or adaptive thresholding can be used. The paper discusses two variants of local thresholding: simple method which is based on the mean and Sauvola's method based on the mean and standard deviation. The input image is divided into square windows (9×9 pixels). For each window a threshold is then determined and thresholding is performed. As opposed to regular thresholding, local thresholding is an example of a contextual operation.

Previous studies on local methods for degraded document images[5] shows that Sauvola performs better than others. That is the proposed algorithm went over large testsutilizing test image databases having textual, pictorialand synthetically generated document images withground-truths and degradations. The results show especially good adaptation into different defect types such asillumination, noise and resolution changes. The algorithmshowed robust behaviour in most, even severe,situations in degradation and performed well against thecomparison techniques[3]. However this method is sensitive to noise from internal circuits and consumes high computation time.

Stochastic computing offers a solution for the fault tolerance and high computation time of Sauvola local image thresholding. Stochastic computing (SC) was as a low-cost alternative to conventional binary computing. It is unique in the sense that it represents and processes information in the form of digitized probabilities. While considering the computation on stochastic bit streams,logical computations are performed on values encoded in randomly streaming bits. This technique can gracefullytolerate high levels of errors. Furthermore, complex operations can be performed with remarkably simple hardware.

III. PROPOSED METHOD

A. Sauvola local image thresholding algorithm

Sauvola's method, take a gray scale image as input. Since most of the document images are colour images, there is a requirement of converting colour image to gray scale image. From the gray scale image, the threshold value at each pixel is computed using Sauvola's method and is given by

$$T = m * \left[1 + k * \left(\frac{s}{R} - 1 \right) \right] \quad (1)$$

where k is a user defined parameter, which is a constant value, m and s are the mean and local standard deviation .respectively which are computed in a window size of w centered on the current pixel. R is the dynamic range of standard deviation. For 8 bit gray scale images R=128.

The relatively well performance on noisy and blurred documents and the computational efficiency are the main advantages of Sauvola's method. In this representation, black is represented by a value 1, and white by a value 0. Conventional Sauvola's method consists of just one step, and its operation is governed by only three parameters. In spite of its advantages, Sauvola's method suffers from the high computational complexity needed to calculate the required statistics, especially for big values of s. So as to overcome this computational complexity and improve the fault tolerance to internal circuits stochastic computing can be used.

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 7, Issue 5, May 2018

B. Stochastic Computation

In stochastic computing, computations in the deterministic Boolean domain are converted into probabilistic computations in the real domain. Two coding formats are used in stochastic computing, the unipolar coding and bipolar coding. In the unipolar coding format, a real number x in the unit interval $0 \leq x \leq 1$ corresponds to a bit stream $X(t)$ of length L , where $t = 1, 2, \dots, L$. The probability that each bit in the stream is 1 is given by $P(X = 1) = x$. In the bipolar coding format, the range of a real number x is extended to $-1 \leq x \leq 1$. The probability that each bit in the stream is 1 is given by $P(X = 1) = (x + 1) / 2$. The two coding formats are the same in essence, and can coexist in a single system. The main trade-off between these two coding formats is that the bipolar format can deal with negative numbers while, given the same bit stream length, L . The precision of the unipolar format is twice that of the bipolar format. To convert a digital value x to a stochastic bit stream X , or to convert the stochastic bit stream back to the corresponding digital value, we need interface circuits. The computational elements in stochastic computing can be implemented with combinational logic or sequential logic or a combination of both [4].

In stochastic computing multiplication can be performed by using an AND gate, subtraction using an XOR gate, mean using a multiplexer and so on. That is computational complexity reduces to a greater extent.

IV. STOCHASTIC IMPLEMENTATION

Thresholding mainly deal with gray scale images. In stochastic implementation we need to scale down the pixel intensities from $[0, 255]$ to $[0, 1]$ interval. While scaling, the constant R in the Sauvola algorithm changes to a value 1. The change in R value modifies the Sauvola algorithm for stochastic implementation as follows

$$T = m * [1 + 0.5(S - 1)] = m * (S + 1) / 2 \tag{2}$$

The processing of the pixel values involves three steps mainly. As stochastic computing mainly deal with random bit streams, first of all we need to convert the pixel values to stochastic bit streams or random bit stream. For that, we can use a random number generator, here we are making use of linear feedback shift registers (LFSR).

The second step involves the generation of threshold bit streams. The window size is one of the most important parameters that depend on Sauvola's algorithm. In this paper we are choosing a window size of 9×9 and so we need to calculate the mean and standard deviation of 81 streams. The mean is calculated using a stochastic mean circuit (SMC) of size 81-to-1.

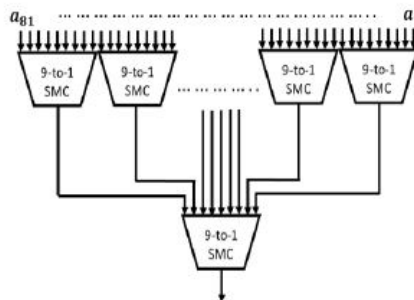


Fig.1 Proposed 81-to-1 SMC

Fig. 1 shows the proposed 81-to-1 SMC. The 81-to-1 SMC can be implemented using ten 9-to-1 SMCs. So as to implement a 9-to-1 SMC, we make use of a 16-to-1 mux whose first eight inputs are connected to 8 uncorrelated input streams and the remaining eight inputs are connected with the eight copies of the ninth bit stream.

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 7, Issue 5, May 2018

Fig.2 shows the simplified block diagram of the generation of threshold bit streams stochastically. The mean value is found out using the SMC shown in Fig.1. The standard deviation of the bit streams are found out using the average value of squares and the square of average values. In order to obtain the squares in stochastic computing, we are using AND gate. But the problem is that, simply using an AND gate alone cannot produce a squared output so before the uncorrelated versions being fed to the AND gate the stream is shifted to one or a few bits. After generation of the squared averages and the average of squares its difference is taken using an XOR gate. The output of the XOR gate is then fed to a stochastic square root circuit and is further scaled addition for generating the threshold bit stream. The process undertaken is simply detailed in the block wise representation shown in Fig.2.

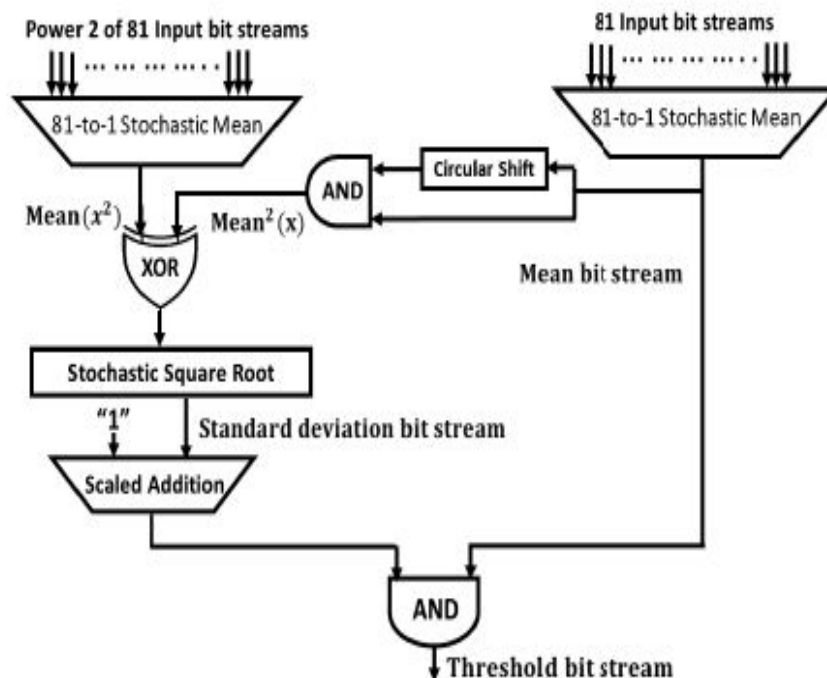


Fig.2 Block wise representation of threshold bit stream generation

Third step is generating the output binary values, it is done using a stochastic comparator. The stochastic comparator we are using is based on a simple counter. The choice of counter is based on the length of input bit streams. The two streams for eg. A and B are compared, by starting the first bit of stream A and if it is 1 we increase the counter value by 1 and continue this for all bits of A. Then we go for bit stream B, if the first bit of second bit stream is 1 and if counter is not empty then the counter value is decreased by 1. By this process finally we can find out which of the two bit streams contains more number of 1s and which is greater than the other. After processing the two streams, if the counter value is zero then the comparator output will be zero else the comparator output will be 1[1].

V. EXPERIMENTAL RESULTS

Figures show the Matlab results of various thresholding methods such as Otsu's method(global), local methods such as Sauvola's, Niblack's and Percentile method. Figure (a) shows the original image and figure (b) shows the collage of results of the methods mentioned. From these results it is clear that why we go for Sauvola's method of local image thresholding. From figure (b) we can see that the output result of Sauvola method (top right most one) is better compared to the other methods, such as Otsu (global), Niblack, and Percentile method.

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 7, Issue 5, May 2018



Fig.3 (a) Original image

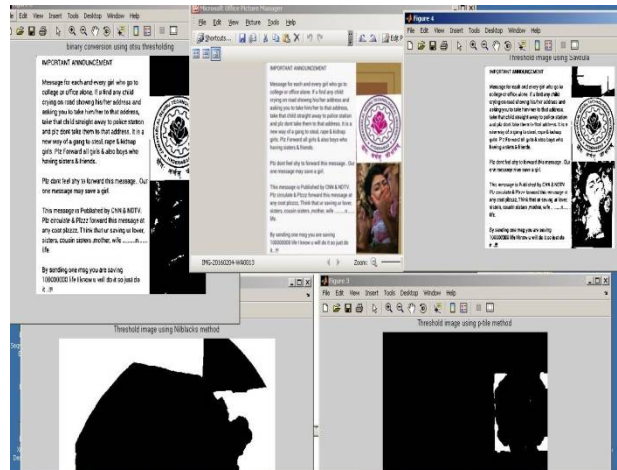


Fig.3 (b) Result collage of different

Stochastic implementation summary using FPGA Spartan6 board:

Device utilization summary

Selected Device : 6slx9tqg144-3

Slice Logic Utilization:

Number of Slice Registers: 26 out of 11440 0%
Number of Slice LUTs: 22 out of 5720 0%
Number used as Logic: 22 out of 5720 0%

Slice Logic Distribution:

Number of LUT Flip Flop pairs used: 27
Number with an unused Flip Flop: 1 out of 27 3%
Number with an unused LUT: 5 out of 27 18%
Number of fully used LUT-FF pairs: 21 out of 27 77%
Number of unique control sets: 2

IO Utilization:

Number of IOs: 3
Number of bonded IOBs: 3 out of 102 2%

Timing Summary:

Speed Grade: -3
Minimum period: 1.683ns (Maximum Frequency: 594.336MHz)
Minimum input arrival time before clock: 2.964ns
Maximum output required time after clock: 5.974ns
Maximum combinational path delay: No path found

VI. CONCLUSION

The stochastic implementation of Sauvola local image thresholding algorithm using a 9*9 window is done. Based on our experimental results, we conclude that stochastic computing systems are extremely tolerant of soft errors. Image processing is a potential application area for SC of great practical importance. Many imaging applications involve functional transformations on the pixels of an input image. The pixel-level functions are usually simple, but because of the large number of pixels involved, the overall transformation process is extremely computation-intensive.

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 7, Issue 5, May 2018

If these functions are implemented using SC, then low-cost, highly parallel image processing becomes possible, as has been demonstrated in a smart SC-based image-sensing chip.

REFERENCES

- [1] M. Hassan Najafi and Mostafa E. Salehi "A Fast Fault-Tolerant Architecture for Sauvola Local Image Thresholding Algorithm Using Stochastic Computing", IEEE Transactions on VLSI systems, vol. 24, no.2, Feb 2016
- [2] Tomasz Kryjaik, Marek Gorgon, "Parallel implementation of local thresholding in Mitron-C", vol.20, no.3, pp.571-580, 2010.
- [3] M. Seznin and B. Sankur, "Survey over Image Thresholding Techniques and Quantitative Performance Evaluation", Journal of Electronic Imaging, vol. 13, No. 1, pp 146-165 (Jan 2004).
- [4] J. Sauvola and M. Pietikäinen, "Adaptive document image binarization," *Pattern Recognit.*, vol. 33, no. 2, pp. 225–236, 2000.
- [5] A. Alaghi and J. P. Hayes, "Survey of stochastic computing," *ACM Trans. Embedded Comput. Syst.*, vol. 12, no. 2s, pp. 1–19, 2013.
- [6] A. Alaghi, C. Li, and J. P. Hayes, "Stochastic circuits for real-time image-processing applications," in *Proc. 50th ACM/EDAC/IEEE Design Autom. Conf. (DAC)*, May/June 2013, pp. 1–6.
- [7] W. Qian, X. Li, M. D. Riedel, K. Bazargan, and D. J. Lilja, "An architecture for fault-tolerant computation with stochastic logic," *IEEE Trans. Comput.*, vol. 60, no. 1, pp. 93–105, Jan. 2011.
- [8] K. Khurshid, I. Siddiqi, C. Faure, and N. Vincent, "Comparison of Niblack inspired binarization methods for ancient documents," *Proc. SPIE, Doc. Recognit. Retr. XVI*, vol. 7247, 2009.
- [9] F. Shafait, D. Keysers, and T. M. Breuel, "Efficient implementation of local adaptive thresholding techniques using integral images," in *Proc. 15th Doc. Recognit. Retr. Conf. (DPR-2008), Part IS&T/SPIE Int. Symp.*
- [10] M. Cheriet, R. F. Moghaddam, and R. Hedjam, "A learning framework for optimization and automation of document binarization methods," An architecture for fault-tolerant computation with stochastic logic," *Comput. Vis. Image Understand.*, vol. 117, no. 3, pp. 269–280, 2013.
- [11] P. Li and D. J. Lilja, "Using stochastic computing to implement digital image processing algorithms," *Proc. IEEE 29th Int. Conf. Comput. Design.*, pp. 154–161, 2011
- [12] P. Li, D. J. Lilja, W. Qian, K. Bazargan and M. D Riedel "Computation on stochastic bit streams digital image processing case studies," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 22, no. 3, pp. 449–462, Mar. 2014.

BIOGRAPHY



Athira Thambi was born in Kerala. She took her B.Tech in Electronics and Communication Engineering from Holy Kings College of Engineering and Technology, Pampakuda (2016). She is doing her M.Tech in VLSI and Embedded System at Viswajyothi College of Engineering & Technology, Muvattupuzha. Her fields of interest include VLSI, Digital System Design and Embedded System.



Merlin Thomas was born in Bahrain. She graduated from the College of Engineering (Adoor), Kerala, India in Electronics & Communication Engineering (2003), completed her M.Tech in Applied Electronics (2012) from Mahatma Gandhi University College of Engineering, Muttom. Currently she is working as Assistant Professor in Viswajyothi College of Engineering & Technology, Vazhakulam, India. Her special fields of interest include VLSI, Time-Frequency Analysis and RF Communication. She is a Life member of the Indian Society for Technical Education. She has 7 Journal & conference papers to her credit.