VLSI IMPLEMENTATION OF HIGH SPEED OPTIMIZED IMAGE SEGMENTATION USING PARALLEL PREFIX ADDER

Ms.M. Praveena [#1], Dr.N. Balaji [#2], Ms.Y. Sai Radhika [#3], Ms.S. Vyshnavi [#4]

#1 ECE Dept., BVRIT Hyderabad College of Engg. for Women, 949249500,
#2 Professor, ECE Dept., JNTUK College of Engg., Narasaraopet, 9848115663,
#3 ECE Dept., BVRIT Hyderabad College of Engg. for Women, 8897240689,
#4 ECE Dept., BVRIT Hyderabad College of Engg. for Women, 7702540244,

ABSTRACT

Genetic algorithms (GA) come under a category of evolutionary algorithms where the elements of search space are binary strings or arrays of other elementary types. The roots of GAs go back to 1950s where they are applied to computer aided simulations to gain an in depth understanding of genetic processes and natural evolution and selection. Recently so many fields of research are utilizing the concepts of GA. One of the popular fields is image processing which is a fast growing field with many applications. The most important part of image processing is image segmentation. In this paper we implement an image segmentation algorithm using optimized threshold generated by GA. We tried to improve the speed of the segmentation process by utilizing parallel prefix adder architectures. The entire design is coded using Verilog and implemented on Virtex 4 FPGA.

Index terms: Genetic Algorithm, Image segmentation, Parallel Prefix Adder, FPGA.

INTRODUCTION

Image Segmentation

Digital image processing has a broad range of applications such as remote sensing, image and data storage for transmission in business applications, medical imaging, acoustic imaging, Forensic sciences and industrial automation. In medical applications such as processing of X-Rays, Ultrasonic scanning, Electron micrographs, Magnetic Resonance Imaging, Nuclear Magnetic Resonance Imaging, etc. Images are very important for diagnosis and treatment. The visual clarity of these images help the doctor diagnose the human physical condition without having to cut them open. Image segmentation is the heart of image processing. Segmentation is the process of partitioning a digital image into multiple segments. The goal of segmentation is to simplify or change the representation of image into something more meaningful and easier to analyze. The quality of segmentation determines how easy it is for the doctor to diagnose. There are different methods for segmenting an image like Thresholding, Clustering, Compression based, Histogram based, Edge detection, Region

growing methods, Graph partitioning, etc,. Thresholding methods are the simplest for segmenting an image. These methods divide the image pixels with respect to their intensity level. These methods are used over images having lighter objects than background i.e. medical images.

Genetic Algorithm

A **GA** is a metaheuristic inspired by the process of natural selection that belongs to the larger class of evolutionary algorithms. Genetic algorithms are commonly used to generate high-quality solutionsto optimization and search problems by relying on bio-inspired operators such as mutation, crossover and selection. Because of the close relation to biology and since genetic algorithms were originally applied to single-objective optimization, the objective functions are often referred to as fitness functions. The most exciting characteristic of GA is its high efficiency for difficult search problems without being stuck in local extreme. Some works applied GA to image segmentation. As segmentation can be seen as a process which finds out the optimal regions partition of an image according to a criterion, GA are well adapted to achieve this goal.

Genetic Algorithms have mainly 3 parameters. i.e. Crossover probability, Mutation probability and Population size. Crossover Probability says how often will the crossover be performed. If there is no crossover, offspring is exact copy of parents. If there is a crossover, offspring is made from parts of parents' chromosome. If crossover probability is 100%, then all offspring is made by crossover. If it is 0%, whole new generation is made from exact copies of chromosomes from old population. Mutation probability says how often will be parts of chromosome mutated. If there is no mutation, offspring is taken after crossover without any change. If mutation is performed, part of chromosome is changed. If mutation probability is 100%, whole chromosome is changed, if it is 0%, nothing is changed. Population size says how many chromosomes are there in a population of one generation. If there are very small chromosomes, GA has few possibilities to perform crossover and only a small part of search space is explored. On the other hand, if there are too many chromosomes, GA slows down.
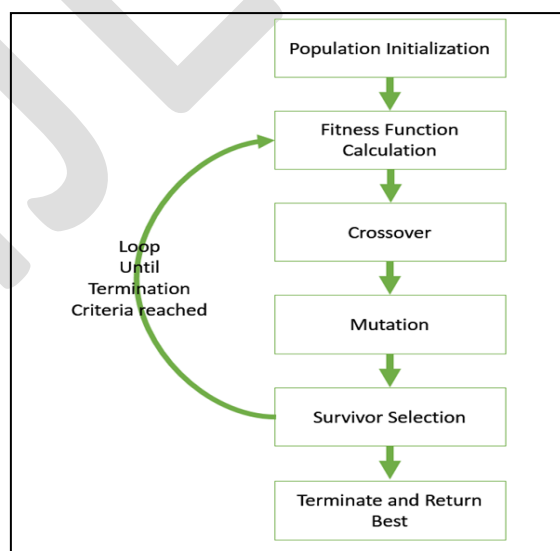

Fig 1: Process of Genetic Algorithm

There are different types of GAs like standard genetic algorithm (SGA), parallel genetic algorithm (PGA) and adaptive genetic algorithm (AGA). Here adaptive genetic algorithm is

used since it improves the speed of convergence. In adaptive genetic algorithm the parameters are adaptive. Instead of using fixed values of probabilities AGAs utilize the population information in each generation and adaptively adjust the probability of crossover and mutation in order to maintain the population diversity as well as to sustain the convergence capacity. In AGA the adjustment of probabilities depends on the fitness values of the solutions.

Parallel Prefix Adder

The name implies that prefix indicates that the outcome of the operation depends on initial inputs. Parallel prefix adders (PPA) are also called as carry tree adders with respect to their structures. PPAs are based on generation and propagation of the carry signals. The process of adding is done in three stages.

- Pre-processing stage
- Carry generation network
- Post processing stage

In the pre-processing stage the generate and propagate signals are computed for given inputs by using basic equations. In the carry generation or prefix stage the group generate and group propagate signals are computed with the help of two operators called black cell and gray cell.

$$G_i = A_i \cdot B_i \ , \ P_i = A_i \oplus B_i$$

$$S_i = P_i \oplus C_i$$

$$C_{i+1} = G_i + P_i \cdot C_i$$

Black Cell

It receives generate and propagate signals from previous level and compute required generate and propagate signals for the next stage.

Gray Cell

It receives generate and propagate signals and compute only the generate signal for the next level.

The group generate and group propagate signals are calculated in intermediate stage to avoid waiting for a ripple. In the post processing or final stage the sum is calculated with the help of the equation.
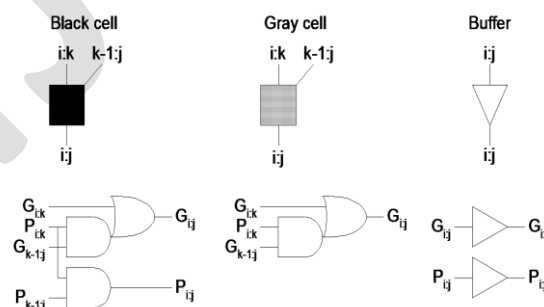
$$G_{i:j} = G_{i:k} + P_{i:k} \cdot G_{k-1:j}$$

$$P_{i:j} = P_{i:k} \cdot P_{k-1:j}$$

Based on the different arrangement of black cells and gray cells i.e. the tree structure, different types of PPAs can be formed. Here the following 4 architectures are used. They are

- Koggestone
- Brent-Kung
- Ladner Fischer
- Sklansky

Related Work

Sudheer Kumar Yezerla et al.[1] did research on, "Design and Estimation of delay, power and area for Parallel Prefix Adders", where the design was implemented using Xilinx Virtex 5 FPGA and the adder delays were estimated using Agilent 1692A Logic Analyzer. The 16-bit SKA computes the carries with Black Cells (BC's) and Grey Cells (GC's) and terminates with a 4-bit RCA.

Senthilkumaran, et al. [2009] [2] has discussed the key goal is toward analysis the concept of edge detection aimed at image segmentation by soft computing method established on the Fuzzy logic, Genetic Algorithm and Neural Network.

Shanta H Biradar[3] has proposed an improved image segmentation algorithm based on improved genetic algorithm which improves the quality of the segmentation for a colour image.

Implementation

In this paper we have designed a high speed optimized GA. In the proposed image segmentation process GA with FPGA implementation is designed by using parallel prefix adder(PPA) instead of the ordinary adder unit. This new adder depending on which tree structure is chosen will either occupy less area or works at more speed. The working of the proposed system is as follows.

1. The initialization module performs population initialization i.e. generates new individuals.
2. The generated individuals are stored in memory.
3. After initialization the fitness values are calculated for each and every individual and stored in memory.
4. If the fitness values are not satisfactory we generate new individuals by using mutation and crossover operations.
5. This process is repeated until we get the optimized threshold value and then this threshold value is provided to the segmentation algorithm.
6. The input image is then segmented using this threshold value.

The main idea of this system is instead of using ordinary adder unit a PPA can be used. This adder can achieve fast arithmetic operation than possible with a normal adder. This adder is mainly used for reducing area and enhancing the speed performance. The designed adder is used in the address generation unit of the GA. This GA module is used to generate optimum threshold values to perform image segmentation.

Experimental Results

The entire architecture is coded in Verilog and implemented on Virtex 4 FPGA with a speed grade of -10. A 14 bit adder using 4 different PPA architectures i.e., Sklansky, Brent Kung, Ladner Fischer, Kogge Stone and also a Carry Select Adder (CSA) was implemented for comparison purpose . All these adder structures have been coded and implemented on Virtex 4 FPGA. The results show that Kogge Stone adder has demonstrated a critical path delay of 10.425ns but occupies more area and Brent Kung adder occupies less area compared to all the other adders i.e., 19 slices and 27 LUTs but has more delay. Sklansky adder offers less delay and occupies less area comparitively. This can be observed from Table 1.

Table 1: Comparison of different adder structures

Adder	Slices	LUT'S	Path delay(ns)	Power(W)		
				Total	Dynamic	Quiescent
SKLANSKY	25	43	12.448	0.166	0	0.166
LADNER FISCHER	21	33	13.690	0.166	0	0.166
BRENT KUNG	19	27	15.223	0.166	0	0.166
KOGGESTONE	45	79	10.425	0.166	0	0.166
CARRY SELECT	28	49	14.547	0.166	0	0.166

After implementing all the 5 adder architectures on Virtex 4 FPGA, these adders are used in the address generation unit of GA which is further used for generating threshold values for segmenting the image.

The simulation waveforms, RTL schematic and Xilinx implementation of a Sklansky adder is shown in the Fig 2, 3 & 4 respectively.
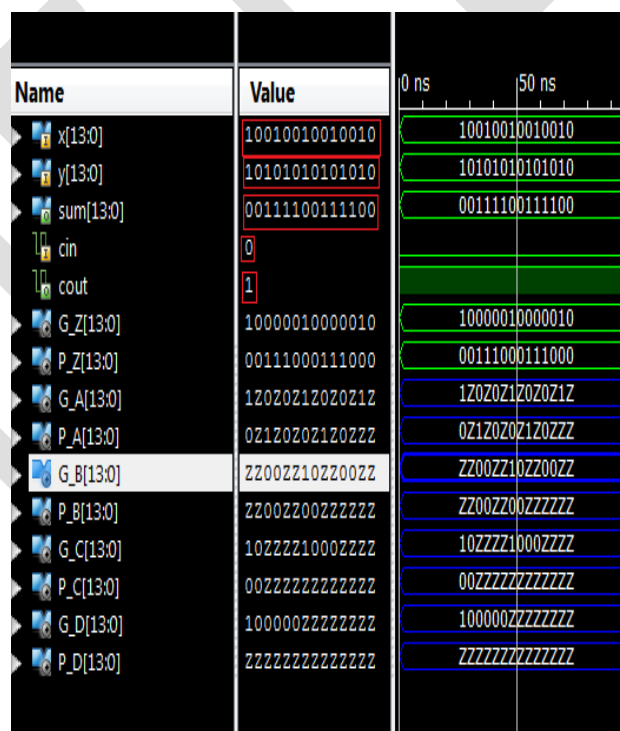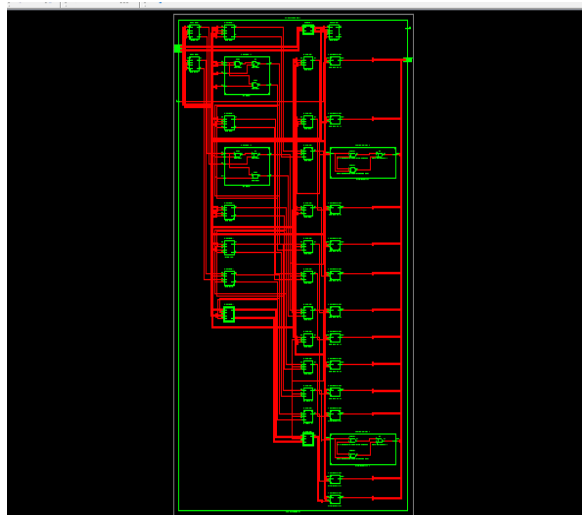

Fig 2: 14 bit Sklansky adder simulation waveforms

Fig 3: RTL Schematic of 14 bit Sklansky adder

The RTL schematic and FPGA implementation of image segmentation algorithm using the above mentioned Sklansky adder is shown in Fig 5 & 6 respectively.
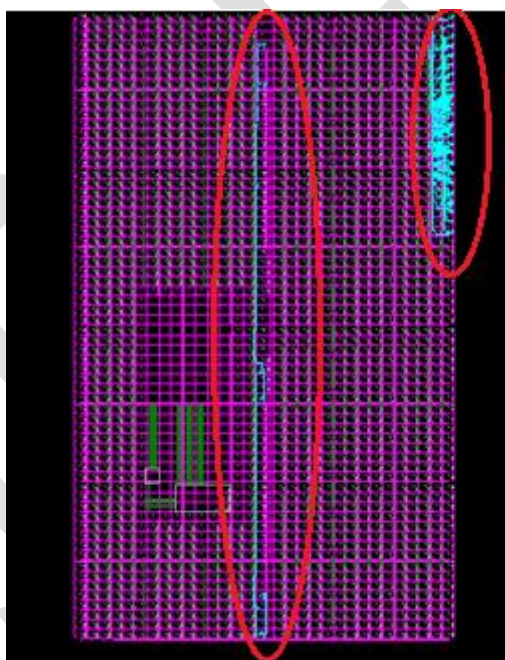

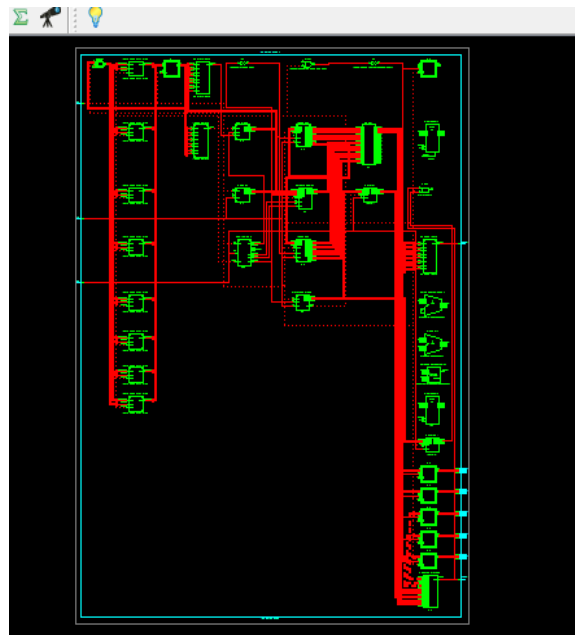Fig 4: FPGA Editor View of 14 bit Sklansky adder

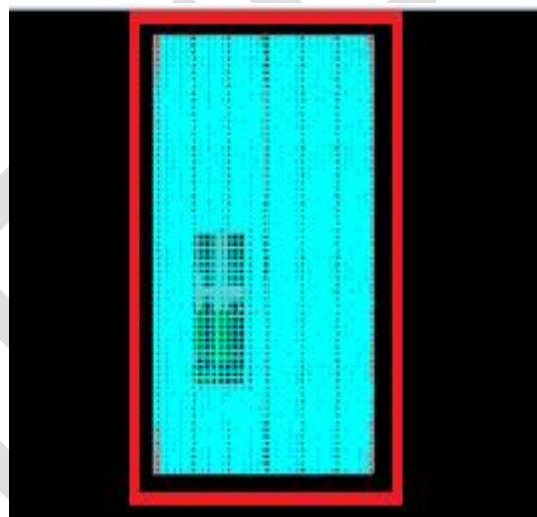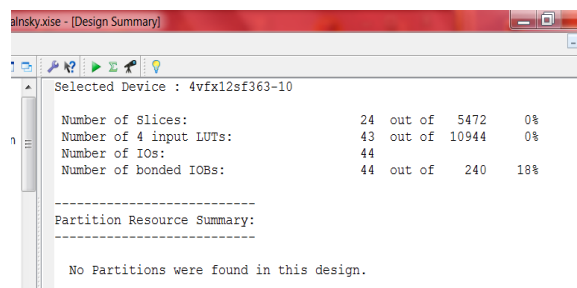Fig 5: RTL schematic of image segmentation

Fig 6: FPGA Editor View of Image Segmentation

The design and timing summary of both these designs are provided here in Fig 7 to Fig 12.

Fig 7: Design Summary of Sklansky adder

```
--------------------------------------
No asynchronous control signals found in this design

Timing Summary:
---------------
Speed Grade: -10

   Minimum period: No path found
   Minimum input arrival time before clock: No path found
   Maximum output required time after clock: No path found
   Maximum combinational path delay: 12.448ns

Timing Detail:
--------------
All values displayed in nanoseconds (ns)
```
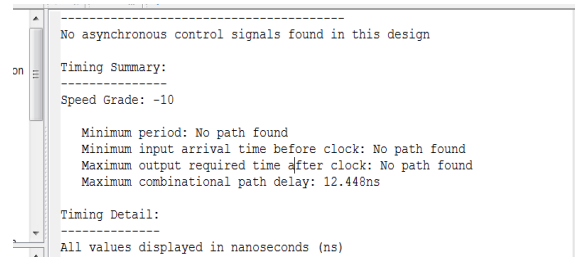
Fig 8: Timing Summary of Sklansky adder
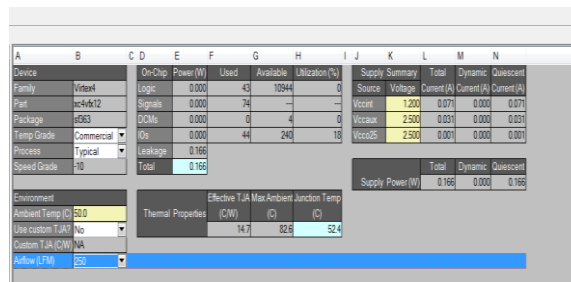
Fig 9: Power Utilization Summary of Sklansky adder

```
Device utilization summary:
---------------------------

Selected Device : 4vfx12sf363-10

   Number of Slices:                    4632  out of   5472    84%
   Number of Slice Flip Flops:           217  out of  10944     1%
   Number of 4 input LUTs:              8603  out of  10944    78%
   Number of IOs:                         30
   Number of bonded IOBs:                 30  out of    240    12%
   Number of FIFO16/RAMB16s:               1  out of     36     2%
      Number used as RAMB16s:              1
   Number of GCLKs:                        1  out of     32     3%

---------------------------
```
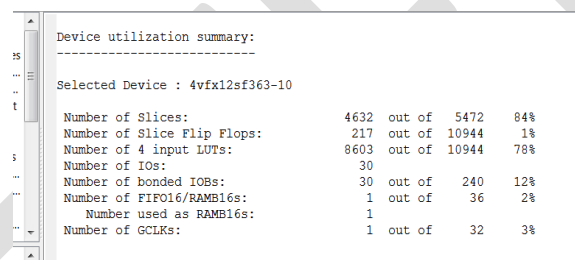
Fig 10: Design Summary of Image Segmentation

```
Timing Summary:
---------------
Speed Grade: -10

   Minimum period: 13.348ns (Maximum Frequency: 74.916MHz)
   Minimum input arrival time before clock: 3.910ns
   Maximum output required time after clock: 12.829ns
   Maximum combinational path delay: No path found

Timing Detail:
--------------
All values displayed in nanoseconds (ns)
```
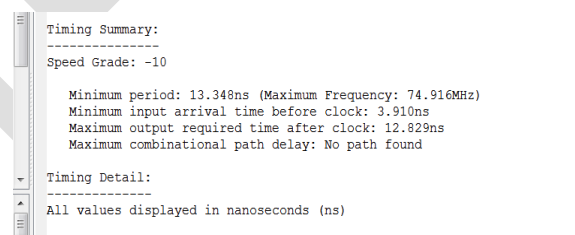
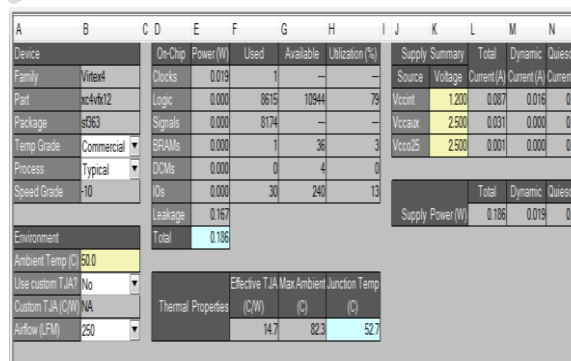Fig 11: Timing Summary of Image Segmentation

Fig 12: Power Utilization Summary of Image Segmentation

This image segmentation algorithm has been implemented with all the 5 adder structures mentioned above and comparing the results show that all PPAs provides less delay compared to CSA and occupy lesser area. The comparison is given in Table 2.

Table 2: Comparison of Image Segmentation using GA with various adders.

Adder	Slices	LUT'S	Frequency(MHz)	Power(W)		
				Total	Dynamic	Quiescent
SKLANSKY	4632	8603	74.916	0.186	0.019	0.167
LADNER FISCHER	4631	8591	73.373	0.189	0.022	0.167
BRENT KUNG	4633	8593	71.674	0.189	0.022	0.167
KOGGESTONE	4643	8626	70.724	0.189	0.022	0.167
CARRY SELECT	4637	8605	67.395	0.189	0.022	0.167

CONCLUSION

This paper introduces high speed architecture for Image segmentation. This design is endowed with a new address generation unit which operates at high speed due to its modified architecture. PPA operates at high speed and occupy less area compared to CSA. In the PPAs koggestone adder can be treated as the high speed adder but it occupies more area compared to other prefix adders. The Brent-kung adder occupies least area and has less speed whereas the sklansky and the ladner-fischer adders are moderate in terms of speed and area but better compared with CSA.

FUTURE SCOPE

High quality image and video processing has become an important part in many professional and consumer applications. Unfortunately, it often comes with a high performance price. Achieving architectural improvement over the existing designs and improving the computational complexity of all the image processing techniques can result in high speed processors which can be used in applications such as Machines vision system, Medical imaging processing, Smart phone image processing, Industrial inspection etc.. In this paper only image segmentation algorithm was improved, which can also be the case for all other image processing techniques like image enhancement, image restoration, image compression etc..

REFERENCE

[1] Y. Choi, "Parallel Prefix Adder Design", Proc. 17th IEEE Symposium on Computer Arithmetic, pp 90-98, 27th June 2005
[2]Senthilkumaran, N., and R. Rajesh. "Edge detection techniques for image segmentation–a survey of soft computing approaches." International journal of recent trends in engineering 1, no. 2 (2009).

[3] Shanta H Biradar "Reboost Image Segmentation Using Genetic Algorithm", International Journal of Computer Science Trends and Technology (IJCST) – Volume 3 Issue 3, May-June 2015

[4] R. P. Brent and H. T. Kung, "A regular layout for parallel adders", IEEE trans, computers, Vol.C- 31,pp. 260-264,.March 1982.

[5] Y. Choi, "Parallel Prefix Adder Design", Proc. 17th IEEE Symposium on Computer Arithmetic, pp 90-98, 27th June 2005

[6] G.L.A.Santhi and G.Deepika, "Realization of Parallel Prefix Adders for Power and Speed Critical Applications", IJVDCSS, Vol.04, pp0118-0123, February-2016.

[7] K.S. Tang*, K.F. Man, Y.C. Ho, S. Kwong "A realizable architecture for genetic algorithm parallelism", Control Engineering Practice 6 (1998) 897 Ð 903.

[8] Gurwant kaur koonar, "A Reconfigurable hardware implementation of Genetic algorithms for vlsi cad design", A Thesis Presented to The Faculty of Graduate Studies of The University of Guelph.

[9] Vijai Singh, Prof.A.K.Mishra, Varsha, "Cardiac Image Segmentation using Simulated Genetic Algorithm", 2015 International Conference on Advances in Computer Engineering and Applications (ICACEA) IMS Engineering College, Ghaziabad, India. 978-1-4673-6911-4/15/$31.00©2015 IEEE

[10] Songhua Xie, Hui Nie, "Retinal Vascular Image Segmentation Using genetic algorithm Plus FCM

clustering", 2013 Third International Conference on Intelligent System Design and Engineering Applications.

Authors

Ms. M. Praveena obtained her B.Tech degree from JNT University in 2003. She received her M.Tech degree from JNT University, Hyderabad in 2007. She is pursuing her Ph.D from JNTU University, Hyderabad in the field of image processing and VLSI implementation. She has authored 2 research papers in International Journals. She is an IEEE member and a life member of IETE and ISTE. Her areas of interest include VLSI, Image Processing and Optimization techniques.

Dr. N. Balaji obtained his B.E degree from Andhra University. He received Master's and PhD degree from Osmania University, Hyderabad. Presently he is working as a professor in the Department of ECE, JNTUK University college of Engineering, Narasaraopet, Andhra Pradesh. He has authored more than 30 Research papers in National and International Conferences and Journals. He is a life Member of ISTE and Member of VLSI Society of India. His areas of research interest are VLSI, Signal Processing, Radar, and Embedded Systems.

Ms. Y. Sai Radhika is pursuing her B.Tech degree in Department of Electronics and Communication Engineering, BVRIT HYDERABAD College of Engineering for Women, Hyderabad, India. She is a member of ISTE & IETE. Her areas of interest include Communications and VLSI.

Ms. S. Vyshnavi is pursuing her B.Tech degree in Department of Electronics and Communication Engineering, BVRIT HYDERABAD College of Engineering for Women, Hyderabad, India. She is a member of ISTE & IETE. Her areas of interest include VLSI & Cellular & Mobile Communications.