SPAA Aware Error Acceptable 2 D Gaussian Smooth Filter for De-Noising of Edge Detection Application

[1] Suhela Malik, [2]Nitin jain
[1]Mtech Student, [2] Assistant Professor
[1]Digital Electronics, [1]Electronics and Telecommunication
[1]Chouksey Engineering College, Bilaspur India

ABSTRACT: In present era every multimedia device are require fast and good quality image/video. Due to Internet of Things there is rapid demand of real time applications so for those applications there is need of some application specific processing unit which also make justice with batter power consumption. As we know in present stage ever one doing real time image/video transmission. Due to real time may be some time there is image quality will decrease so for improvement of those real time image there is need of De-noising approach which I well known as Smooth filter. In this project we will design a fast and error tolerant algorithm which is based on Gaussian smooth filter. Here we also also design our own hardware unit using VLSI technology which is based on powerful HDL "verilog". Our main motto is make justice with SPAA(Speed, Power, Area Accuracy) Metrics. For application analysis we will use Edge detection because here we have to verify the quality level of our proposed smooth filter. For quality analysis we will use Image quality parameters like PSNR, SSIM, FSIM, RFSIM, GMSD. I will compare my proposed algorithm & architecture with previous existing approach. Implementation of proposed algorithm will be done by Matlab and hardware implementation will be done by using of Verilog on Xilinx 14.1 simulator. Verification will be done on Modelsim.

Keywords ***ASAP, PSNR, SSIM, RFSIM, FSIM, GMSD, HDL***

I. INTRODUCTION

II. In current sciences and innovations, pictures additionally increase substantially more extensive degrees because of the regularly developing significance of logical perception (of frequently vast scale complex logical/trial information). Illustrations incorporate microarray information in hereditary research, or ongoing multi-resource portfolio exchanging money. Before going to preparing a picture, it is changed over into a computerized frame. Digitization incorporates examining of picture and quantization of inspected esteems. In the wake of changing over the picture into bit data, preparing is performed. This handling procedure might be Image improvement, Image rebuilding, and Image compression.[4] Image upgrade: It alludes to highlight, or honing, of picture highlights, for example, limits, or differentiation to make a realistic show more helpful for show and investigation. This procedure does not expand the inalienable data content in information. It incorporates dim level and complexity control, clamor decrease, edge crispening and honing, sifting, introduction and amplification, pseudo shading, et cetera. Picture rebuilding: It is worried about sifting the watched picture to limit the impact of debasements. Viability of picture reclamation relies upon the degree and exactness of the learning of debasement process and also on channel plan. Picture reclamation contrasts from picture improvement in that the last is worried about more extraction or highlight of picture highlights. Picture pressure: It is worried about limiting the quantity of bits required to speak to a picture. Use of pressure are in communicated TV, remote detecting through satellite, military correspondence by means of air ship, radar, video chatting, copy transmission, for instructive and business reports, therapeutic pictures that emerge in PC tomography, attractive reverberation imaging and computerized radiology, movement, pictures, satellite pictures, climate maps, topographical studies et cetera. Picture preparing is characterized as the control of picture portrayal put away on a PC. Tasks on pictures that are viewed as a type of picture preparing incorporate zooming, changing over to dim scale, expanding or diminishing picture splendor, red-eye decrease in photos, edge and shape recognition of a question and investigation of protest properties, for example, size and shading. These tasks normally include cycle over every single individual pixel in a picture.

III. In present era energy consumption has become critical issue for multimedia devices like mobile and embedded systems [1]. These devices incorporate computer graphics and image pro- cessing as their core application like security, medical science, entertainment, etc. The real time image processing suffers from noise signal which degrades the quality of the image while compression or transmission. For reduction of these noise, there is a need of smooth filter. Commercially smooth filters are available having characteristic like Averaging, Median, Mean, Gaussian Filter etc. Out of these, the mainly used filter is a 2D Gaussian Smooth Filter (2GSF), since it does not generate false edge on increasing scale [2] and provides good trade-off between localization in spatial and frequency domains. The famous edge detection algorithms like Canny [3] and MarrHildreth [4] uses 2GSF. 2GSF is also useful in many other applications like texture segmentation [5], tone mapping of high dynamic range images

[6], image blurring [7] and image mosaicing [8]. 2GSF is basically based on gaussian kernel function, having a floating point coefficient. In terms of hardware complexity, floating point design requires large hardware unit and huge amount of energy. Existing approaches will make injustice with SPAA (Speed, Power, Area, Accuracy) metrics, while operating on portable devices. In order to justify SPAA metrics there is need of approximate design. The approximate designs produce almost-correct results, and offer power reductions with performance improvements in return [9]. These designs exploit a tradeoff of accuracy explained with an example, let there be two number X=223 and Y=224. Its accurate and approximate multiplication [10] will results to Z = 49952 and Z = 46847. The total and percent error difference is 3105 and 6.21% respectively. As per [11], the human eye can tolerate an error upto 10%. So we can easily apply approximation on image processing system. Through this small error there is tremendous saving in hardware complexity.

II. LITRECTURE REVIEW

Literature study is basically done to understand the gabor filter and different parameter we used to compare different gaussian filter we implement in this paper. Literature study suggest some algorithm which can be implemented to improve the efficiency of gaussian filter.

The fields of image processing and computer vision are continuously gaining increased attention in applications including robotics, automation, quality control, and security systems. Among the many image processing procedures, edge detection is seen by many as the first essential step in any type of image analysis. It is used to separate the image into object(s) and backgr ound. The performance of an edge detection operator is defined as its ability to locate, in noisy data, an edge that is as close as possible to its true position in the image.

- 2D Gaussian Smooth Filter:

2GSF is one of the most common filter which is used in many image processing applications. This filter is based on a fixed value of standard deviation. The equation of a Gaussian function in 2D with a standard deviation can be described by:

$$g(x, y) = e^{\frac{-(x^2+y^2)}{2\sigma^2}}$$

Where x & y is distance from origin of horizontal & vertical axis. When this formula is applied on 2D image, it produces a surface whose contours are concentric circles with a Gaussian distribution from the centre point. When a noisy image t is applied to gaussian filter with impulse response of g, so spatial domain of smooth image k is calculated by using of convolution.

$$k(x, y) = t(x, y)*g(x, y)$$

Similarly, frequency domain of smooth image k is calculated by using of below expression

$$K(a, b) = T(a, b) \times G(a, b)$$

Frequency domain is generated by spatial domain, which are k(x,y), t(x,y) and g(x,y). Here K(a,b), (a,b) and G(a,b) are represent frequency domain form. In equation (1) will decide amount of smoothness. For more effective smoothing large value of σ & large kernel is required for accurate representation of a function. In this paper we have proposed gaussian smooth filter of 3X3 and 5X5 kernel for _=1, which is calculated by equation (1) for given (x,y) values. For calculation of 3 X3 kernel, value of x & y lies between -1 to 1. Coefficients of a 3X3 gaussian kernel for σ= 1 are :

$$\begin{bmatrix} 0.0751 & 0.1238 & 0.0751 \\ 0.1238 & 0.2042 & 0.1238 \\ 0.0751 & 0.1238 & 0.0751 \end{bmatrix}$$

For calculation of 5 X5 kernel, value of x & y lies between -2 to 2. Coefficients of a 5X5 gaussian kernel for σ = 1 are:

$$\begin{bmatrix} 0.0030 & 0.0133 & 0.0219 & 0.0133 & 0.0030 \\ 0.0133 & 0.0596 & 0.0983 & 0.0596 & 0.0133 \\ 0.0219 & 0.0983 & 0.1621 & 0.0983 & 0.0219 \\ 0.0133 & 0.0596 & 0.0983 & 0.0596 & 0.0133 \\ 0.0030 & 0.0133 & 0.0219 & 0.0133 & 0.0030 \end{bmatrix}$$

2D gaussian smooth filter is basically based on gaussian kernel function, having a floating point coefficient. In terms of hardware complexity, floating point design requires large hardware unit and huge amount of energy. Existing approaches will make injustice with ASAP (Speed, Power, Area, Accuracy) metrics, while operating on portable devices. In order to justify ASAP metrics there is need of approximate design.

Approximation For Error Tolerant Applications: The approximate designs produce almost-correct results, and offer power reductions with performance improvements in return. This design exploits a tradeoff of accuracy in computation versus speed, power and area. Explained with an example, let there be two number X=223 and Y=224. Its accurate and approximate multiplication [8] will results to Z = 49952 and Z = 46847. The total and percent error difference is 3105 and 6.21% respectively. As per [7], the human eye can tolerate an error upto 10%. So we can easily apply approximation on image processing system. Through this small error there is tremendous saving in hardware complexity.

Approximate 2D Gaussian Smooth Filter: As we already see there is no need of accurate logic we can use approximate logic and reduce the previous existing issue. In this direction many researchers are make there 2D Gaussian system. Here am targeting some of them. According to [13] author propose a technique which is based on approximation, in this paper author change the Gaussian kernel value and try to reduce the hardware complexity. Here author use power-of-two. According to modified[14] author propose a modified approximate technique which is used for 2D Gaussian smooth filter. Here author use the fixed point technique and modified the Gaussian kernel. According to Ankur[15] author propose a technique which is use to design fast 2D Gaussian smooth filter. According to this approach author use a diagonal technique where he reduce the input image 33 and 5X5 kernel. According to Sharda[16] author modified the [15] technique and reduce the diagonal kernel into semi diagonal kernel. This all technique are able to reduce the harward complexity specially [15] & [16] but in [15] [16] image quality is not so good.

III. METHADOLOGY & IMPLIMENTATIONDETAILS

In this paper we will propose a new algorithm which will make proper justification with SPPA metrics & also approach the good image quality. Here we will use approximation approach for achieve my paper objective. Implementation will be done in both level means algorithm & architecture level. Application analysis will be done on Noisy based Edge detection and all image quality analysis will be done by image quality parameter like PSNR, SSIM, RFSIM, FSIM, GMSD. Here we are following PLUS structure:

Error Acceptable 3X3 Gaussian Smooth Filter

Original Gaussian 3X3 kernal have total 9 coeffiecient and those coefficient are make convolution operation with 3X3 input image matrix an generate smooth output image. But as we already seen the concept of approimation in image/ video application, so there is no need to apply accurate logic but as per the previous esxisting approimate gaussian approximate techniue. They are not capable to make justice with output image quality. So in our case we are focucing on quality also so here we are using plus mask of 3X3 gauusian where we are using total 5 coeffient. As shown in below figure.

0.0751	0.1238	0.0751
0.1238	0.2042	0.1238
0.0751	0.1238	0.0751

Here we select total five coefficient and we convert those original coefficient value in to fixed point which we shown in below.

Gaussian Kernal for 3X3:

First Row: 1/4 =0.25

Second Row: 1/8=0.125, 3/16=0.1875, 1/4= 0.25

Third Row: 1/4 = 0.25

Error Acceptable 5X5 Gaussian Smooth Filter

Simmilar we also use the same logic of plus mask here we are using total 9 coeffients which shown in below figure.

0.0030	0.0133	0.0219	0.0133	0.0030
0.0133	0.0596	0.0983	0.0596	0.0133
0.0219	0.0983	0.1621	0.0983	0.0219
0.0133	0.0596	0.0983	0.0596	0.0133
0.0030	0.0133	0.0219	0.0133	0.0030

Here we select total nine coefficient and we convert those original coefficient value in to fixed point which we shown in below.

Gaussian Kernal for 5X5:

First Row:7/128=0.0546

Sencond Row:1/2=0.25

Third Row: 5/256=0.0195, 3/32=0.09375, 5/32=0.1562, , 3/32=0.09375, 5/256=0.0195

Fourth Row:1/2=0.25

Fifth Row: 7/128=0.0546

Implementation of Accurate 3X3 & 5X5: According to this approach basically we are using total 9 coefficients for 3X3 and 25 pixels for 5X5. So on both case there is hardware and time complexity increase which we shown in result analysis section. But this approach is best in terms of quality.

Implementation of Modified 3X3 & 5X5: This approach is implemented by using of matlab. According to this approach basically author using total 9 coefficients for 3X3 and 25 coefficients for 5X5. But here one important improvement they make , they use fixed point concept so every pixel are easily to make multiplication operation by using of left shifting operation. But still there is issue of is hardware and time complexity increase which we shown in result analysis section.

Implementation of Ankur 3X3 & 5X5: This approach is implemented by using of matlab. According to this approach basically author using diagonal mask. Here they use 3 coefficients for 3X3 and 5 coefficients for 5X5. They also use the concept of fixed point so every pixel are easily to make multiplication operation by using of left shifting operation. They got improved architecture in terms of area, power and speed, they loss the quality of output image.

Implementation of Sharada 3X3 & 5X5: This approach is implemented by using of matlab. According to this approach basically author using half diagonal mask. Here they use 2 coefficients for 3X3 and 3 coefficients for 5X5. They also use the concept of fixed point so every pixel are easily to make multiplication operation by using of left shifting operation. They g ot improved architecture in terms of area, power and speed, they loss the quality of output image.

Proposed 2D Gaussian Filter:

Here we will present the gate level and Lut based design which is generated by using of Xilinx 14.2. Here we will implement our plus based logic which will produce a small logic. As compare to previous approach.

Proposed 3X3:

Top Module of our proposed Architecture:

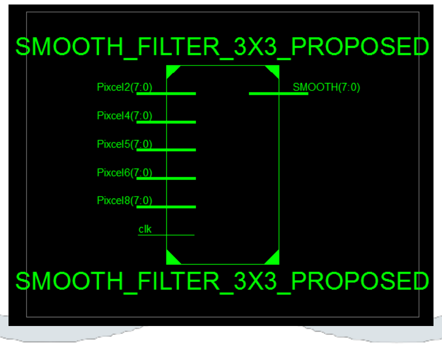


Fig. 3.1 Proposed 3X3 Top Module

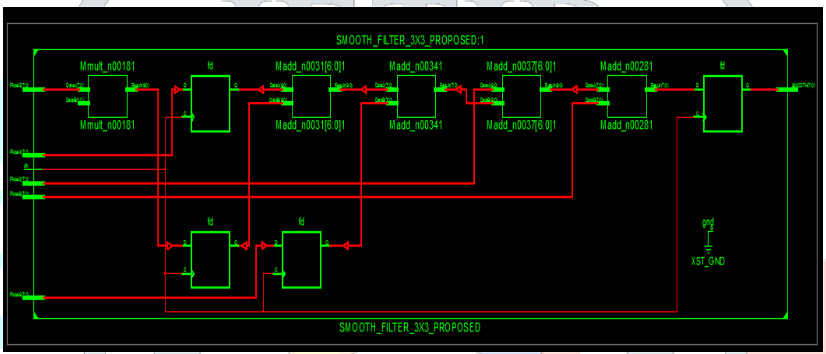


Fig. 3.2 Proposed 3X3 Lut Level Design

Proposed 5X5:

Top Module of our proposed Architecture:

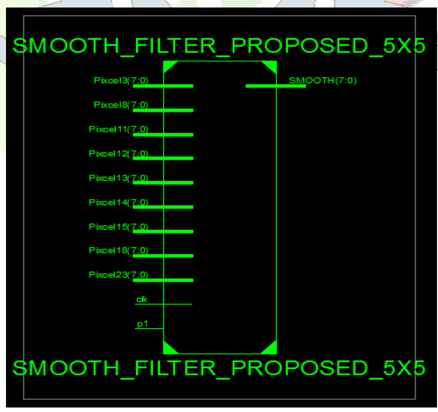


Fig. 3.3 Proposed 5X5 Top Module

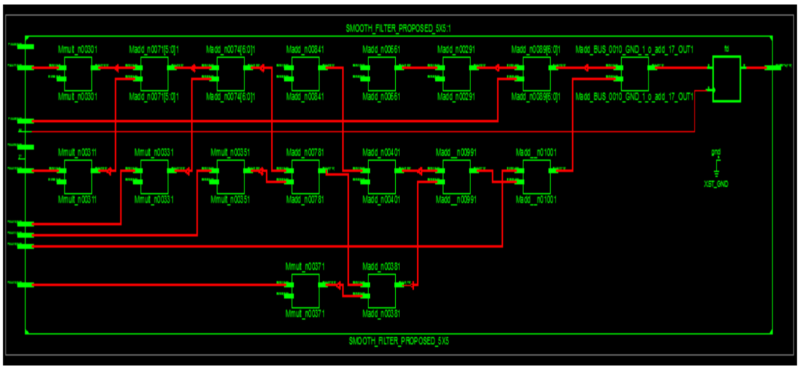


Fig. 3.4 Proposed 5X5 Lut Level Design

IV. RESULT & ANALYSIS

A new algorithm will be proposed and that algorithm will be implemented by using of MATLAB, for image quality measurement I will use some scientific parameters like PSNR, SSIM, FSIM, RFSIM, GMSD, Similarity (%). I will also propose hardware unit for my proposed algorithm which will reduce the area, power and speed problem. I will compare my proposed algorithm with previous approach hardware implementation will be done by using of Verilog on Xilinx 14.2 simulator. Verification will be done on Modelsim. During first stage the proposed method is implemented on matlab to thoroughly investigate the required time to reduce noise within an object and compare output image with various parameters .

Here we are presenting the timing complexity analysis for 3X3 and 5X5gaussian kernel:

Time Complexity Analysis:

3X3 2D Gaussian Smooth Filter:

Parameter	Accurate_3X3	Modified_3X3	Sharda_3X3	Ankur_3X3	Proposed_3X3
Time(Sec)	0.3744	0.2808	0.0624	0.1092	0.1248

5X5 2D Gaussian Smooth Filter:

Parameter	Accurate_5X5	Modified_5X5	Sharda_5X5	Ankur_5X5	Proposed_5X5
Time(Sec)	0.6396	0.3432	0.0936	0.1248	0.1716

As we can see for both kernel our proposed 2D gaussain smooth filter is require less time as compare ro accurate and modified. But in our proposed approach quality is batter than previous existing approach like Ankur and Sharda.

PSNR:

3X3 2D Gaussian Smooth Filter:

IMAGE	Noise	Accurate_3X3	Modified_3X3	Sharda_3X3	Ankur_3X3	Proposed_3X3
Lena	20.02	27.666	26.963	23.471	24.682	26.113

Here as we can see for proposed 3X3 2D Gaussian smooth filter. The output image quality is far batter as compare to sharda an d Ankur approach.

IMAGE	Noise	Accurate_5X5	Modified_5X5	Sharda_5X5	Ankur_5X5	Proposed_5X5
Lena	20.02	26.23	23.55	23.07	24.34	24.61

Here as we can see for proposed 5X5 2D Gaussian smooth filter. The output image quality is far batter as compare to sharda an d Ankur approach.

SSIM:
3X3 2D Gaussian Smooth Filter:

IMAGE	Noise	Accurate_3X3	Modified_3X3	Sharda_3X3	Ankur_3X3	Proposed_3X3
Lena	0.56	0.701	0.689	0.625	0.639	0.667

Here as we can see for proposed 3X3 2D Gaussian smooth filter. The output image quality is far batter as compare to sharda an d Ankur approach.

5X5 2D Gaussian Smooth Filter:

IMAGE	Noise	Accurate_5X5	Modified_5X5	Sharda_5X5	Ankur_5X5	Proposed_5X5
Lena	0.56	0.7347	0.6717	0.613	0.6091	0.6122

Here as we can see for proposed 5X5 2D Gaussian smooth filter. The output image quality is far batter as compare to sharda an d Ankur approach.

FSIM:
3X3 2D Gaussian Smooth Filter:

IMAGE	Noise	Accurate_3X3	Modified_3X3	Sharda_3X3	Ankur_3X3	Proposed_3X3
Lena	0.799	0.8644	0.863	0.843	0.845	0.853

Here as we can see for proposed 3X3 2D Gaussian smooth filter. The output image quality is far batter as compare to sharda an d Ankur approach.

IMAGE	Noise	Accurate_5X5	Modified_5X5	Sharda_5X5	Ankur_5X5	Proposed_5X5
Lena	0.805	0.845	0.841	0.821	0.831	0.8411

Here as we can see for proposed 5X5 2D Gaussian smooth filter. The output image quality is far batter as compare to sharda an d Ankur approach.

RFSIM:
3X3 2D Gaussian Smooth Filter:

IMAGE	Noise	Accurate_3X3	Modified_3X3	Sharda_3X3	Ankur_3X3	Proposed_3X3
Lena	0.25	0.4229	0.3685	0.2931	0.328	0.3556

Here as we can see for proposed 3X3 2D Gaussian smooth filter. The output image quality is far batter as compare to sharda an d Ankur approach.

5X5 2D Gaussian Smooth Filter:

IMAGE	Noise	Accurate_5X5	Modified_5X5	Sharda_5X5	Ankur_5X5	Proposed_5X5
Lena	0.25	0.3995	0.3136	0.2626	0.2619	0.2632

Here as we can see for proposed 5X5 2D Gaussian smooth filter. The output image quality is far batter as compare to sharda an d Ankur approach.

GMSD:
3X3 2D Gaussian Smooth Filter:

IMAGE	Noise	Accurate_3X3	Modified_3X3	Sharda_3X3	Ankur_3X3	Proposed_3X3
Lena	0.8054	0.8607	0.8518	0.8446	0.8445	0.852

Here as we can see for proposed 3X3 2D Gaussian smooth filter. The output image quality is far batter as compare to sharda an d Ankur approach.

5X5 2D Gaussian Smooth Filter:

IMAGE	Noise	Accurate_5X5	Modified_5X5	Sharda_5X5	Ankur_5X5	Proposed_5X5
Lena	0.8054	0.8452	0.8401	0.8486	0.8471	0.8496

Here as we can see for proposed 5X5 2D Gaussian smooth filter. The output image quality is far batter as compare to sharda an d Ankur approach.

Similarity (%):
3X3 2D Gaussian Smooth Filter:

IMAGE	Noise	Accurate_3X3	Modified_3X3	Sharda_3X3	Ankur_3X3	Proposed_3X3
Lena	79.65	91.75	91.17	86.71	88.36	90.18

Here as we can see for proposed 3X3 2D Gaussian smooth filter. The output image quality is far batter as compare to sharda an d Ankur approach.

5X5 2D Gaussian Smooth Filter:

IMAGE	Noise	Accurate_5X5	Modified_5X5	Sharda_5X5	Ankur_5X5	Proposed_5X5
Lena	79.65	91.26	91.21	87.4	88.61	89.34

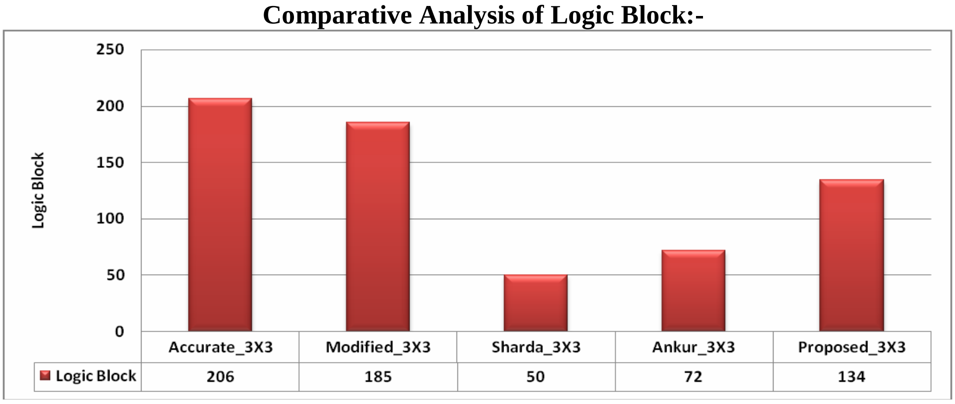
Here as we can see for proposed 5X5 2D Gaussian smooth filter. The output image quality is far batter as compare to sharda an d Ankur approach.

- **Hardware Level Analysis**

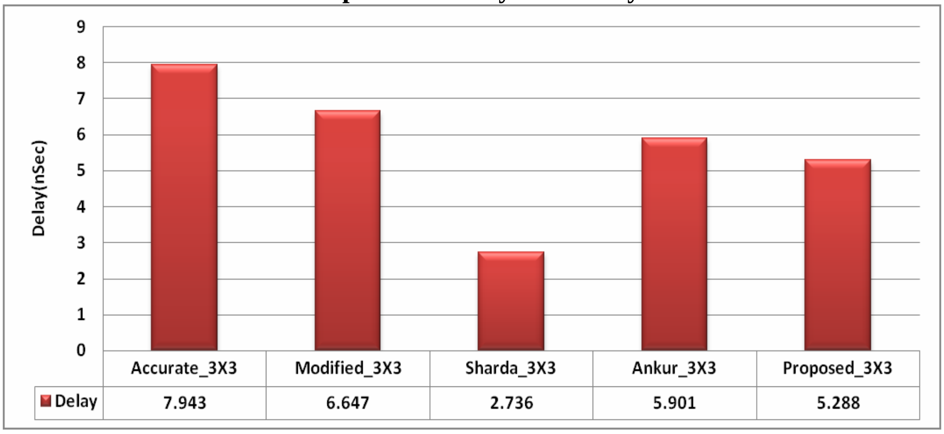
Comparative Result of FPGA :- 3X3 2D Gaussian Filter

PARAMETER	Accurate_3X3	Modified_3X3	Sharda_3X3	Ankur_3X3	Proposed_3X3
Logic Block	**206**	**185**	**50**	**72**	**134**
Delay(nSec)	**7.943**	**6.647**	**2.736**	**5.901**	**5.288**
Frequency(MHz)	**125.89**	**150.44**	**365.49**	**169.46**	**189.1**

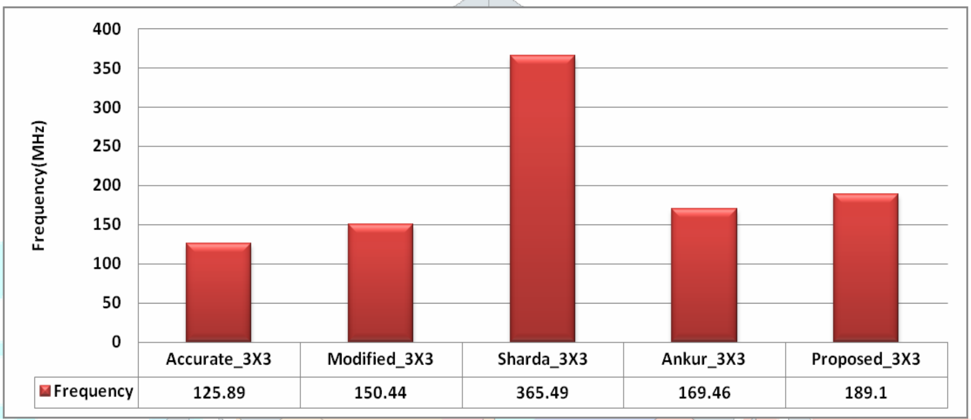
Comparative Analysis of Logic Block:-



Comparative Analysis of Delay:-



Comparative Analysis of Frequency:-



V. CONCLUSION

According to this paper we will resolve the previous existing problems which are latency, power, area. The key contribution of this work is to develop a fast 2D Gaussian algorithm. Using this work we will develop a SPAA aware error tolerant 2D Gaussian system. This proposed unit is design for both 3X3 and 5X5 which require less area, power and speed. In this approach I will propose a new approach of approximation which will reduce some amount of accuracy. In proposed approach I will use only 5 Gaussian coefficient for 3X3 and 9 coefficients for 5X5. Here we are using plus mask logic. We test our proposed algorithm for Noisy Sobel edge detection algorithm. Using this approach I will expect that it will reduce the timing complexity and hardware complexity with 20-30%.

REFERENCES

[1] 1. Prasad, A. V. S. S., Jacob Mathews, and Nagi Naganathan. "Low-power design strategies for mobile computing." *VLSI Design, 2006. Held jointly with 5th International Conference on Embedded Systems and Design., 19th International Conference on*. IEEE, 2006.

2. A. Yuille and T. A. Poggio, “Scaling theorems for zero crossings,” Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. PAMI-8, no. 1, pp. 15–25, Feb 1986.

3. J. Canny, “A computational approach to edge detection,” Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. PAMI-8, no. 6, pp. 679–698, Nov 1986.

4. D. Marr and E. Hildreth., “Theory of edge detection,” Proceedings of the Royal Society of London., vol. 207, p. 1167, Jan 1980.

5. M. Unser and M. Eden, “Multiresolution feature extraction and selection for texture segmentation,” Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 11, no. 7, pp. 717–728, Jul 1989.

6. E. Reinhard, M. Stark, P. Shirley, and J. Ferwerda, “Photographic tone reproduction for digital images,” pp. 267–276, 2002.

7. Guo, Zhengyang, Wenbo Xu, and Zhilei Chai. "Image edge detection based on FPGA." *Distributed Computing and Applications to Business Engineering and Science (DCABES), 2010 Ninth International Symposium on*. IEEE, 2010.

8. Kulkarni, Parag, Puneet Gupta, and Milos Ercegovac. "Trading accuracy for power with an underdesigned multiplier architecture." *VLSI Design (VLSI Design), 2011 24th International Conference on*. IEEE, 2011.

9. Zhu, Ning, Wang Ling Goh, and Kiat Seng Yeo. "An enhanced low-power high-speed adder for error-tolerant application." *Integrated Circuits, ISIC'09. Proceedings of the 2009 12th International Symposium on*. IEEE, 2009.

10. Lee, Kyoungwoo, et al. "Error-exploiting video encoder to extend energy/qos tradeoffs for mobile embedded systems." *Distributed Embedded Systems: Design, Middleware and Resources*. Springer US, 2008. 23-34.

11. Kyaw, Khaing Yin, Wang Ling Goh, and Kiat Seng Yeo. "Low-power high-speed multiplier for error-tolerant application." *2010 IEEE International Conference of Electron Devices and Solid-State Circuits (EDSSC)*. 2010.

12. Kahng, Andrew B., and Seokhyeong Kang. "Accuracy-configurable adder for approximate arithmetic designs." *Proceedings of the 49th Annual Design Automation Conference*. ACM, 2012.

13. Hsiao, Pei-Yung, et al. "A parameterizable digital-approximated 2D Gaussian smoothing filter for edge detection in noisy image." *Circuits and Systems, 2006. ISCAS 2006. Proceedings. 2006 IEEE International Symposium on*. IEEE, 2006.

14. Khorbotly, Sami, and Firas Hassan. "A modified approximation of 2D Gaussian smoothing filters for fixed-point platforms." *System Theory (SSST), 2011 IEEE 43rd Southeastern Symposium on*. IEEE, 2011.

15. Jaiswal, Ankur, et al. "ASAP-Aware 2D Gaussian Smoothing Filter Design Using Efficient Approximation Techniques." *VLSI Design (VLSID), 2015 28th International Conference on*. IEEE, 2015.

16. Kaushik, Sharda, and N. V. S. V. Kumar. "Energy-efficient approximate 2D Gaussian smoothing filter for error tolerant applications." *Advance Computing Conference (IACC), 2015 IEEE International*. IEEE, 2015.

17. Wang, Zhou, et al. "Image quality assessment: from error visibility to structural similarity." *Image Processing, IEEE Transactions on* 13.4 (2004): 600-612.

18. Zhang, Lin, Lei Zhang, and Xuanqin Mou. "RFSIM: A feature based image quality assessment metric using Riesz transforms." *Image Processing (ICIP), 2010 17th IEEE International Conference on*. IEEE, 2010.

19. Zhang, Lin, et al. "FSIM: a feature similarity index for image quality assessment." *Image Processing, IEEE Transactions on* 20.8 (2011): 2378-2386.

20. Xue, Wufeng, et al. "Gradient magnitude similarity deviation: a highly efficient perceptual image quality index." *Image Processing, IEEE Transactions on* 23.2 (2014): 684-695.