Reduction of Power in Adaptive Edge Enhanced Image Scalar Using ETA Adder

D.Santhiya[1], M.Santhanaraj[2]

PG Scholar, Department of Electronics & Communication engineering, KPR Institute of Engineering and Technology, Coimbatore, Chennai, India[1]

Assistant Professor, Department of Electronics &Communication Engineering, KPR Institute of Engineering and Technology, Coimbatore, Chennai, India[2]

Abstract: - **Image scaling is a very important technique and has been widely used in many image processing applications. An adaptive edge-enhanced image scalar adopted in this paper is a low complexity image scaling algorithm. It consists of linear space variant edge detector, a low complexity sharpening spatial filter, and a simplifed bilinear interpolation. The proposed image scaling algorithm uses hardware sharing technique in order to reduce the computational complexity and to minimize the computing resources needed. Furthermore, Error Tolerant Adder(ETA) Unit is used to overcome redundant calculation performed in Reconfigurable Calculation Unit (RCU), which reduce the computational resources and hardware cost required, which further results in reduction of gate count. Error tolerant adder (SPST) adder results in reduction in power consumption and filter out the unused switching power and also increases speed.**

Keywords- 2-D image scalar, edge detector, image zooming, sharpening spatial filter, very large scale integration (VLSI), error tolerance, adders.

I. INTRODUCTION

Image scaling is the process of resizing a digital image scaling is a non trivial process that involves a trade-off between efficiency, smoothness and sharpness. Image scaling is widely used in many fields, ranging from consumer electronics to medical imaging. It is indispensable when the resolution of an image generated by a source device is different from the screen resolution of a target display. For example, we have to enlarge images to fit HDTV or to scale them down to fit the mini-size portable LCD panel. The most simple and widely used scaling methods are the nearest neighbor and bilinear techniques. In recent years, many efficient scaling methods have been proposed in the literature.

Compared to various interpolation methods the simplest interpolation from a computational standpoint is the nearest neighbor, where each interpolated output pixel is assigned the value of the nearest sample point in the input image. However, the images that are scaled by the nearest-neighbor algorithm are full of blocking and aliasing artifacts. Bilinear Interpolation determines the grey level value from the weighted average of the four closest pixels to the specified input coordinates, and assigns that value to the output coordinates. It is widely used due its low complexity, ease of implementation and image quality.

Another non adaptive algorithm is that bicubic interpolation algorithm which is an extension cubic interpolation results in high quality image. However it is computationally expensive due its computational complexity and requires a high memory capacity. Many adaptive interpolation algorithms have been used proposed, such as adaptive 2-D autoregressive modelling, blending kernels, curvature interpolation, and bilateral filter.

Though above image scaling methods produces high quality images but they are not easy to implement with very large scale integration circuits because of computational complexity and high memory requirement. There are many studies concerning the VLSI implementation of low complexity interpolation algorithms. the *winscale* algorithm uses an area pixel model and a domain filtering method for image scaling. each new pixel of the scaled image is obtained by weighted-averaging the pixel values of the original pixels with area coverage ratio. But we cannot obtain good image quality from this algorithm because repetition results in blocky effects and they also require much larger hardware resources.

In an edge-oriented area pixel scaling processor was proposed. It uses a low-cost edge catching technique and seven-stage pipeline architecture to achieve low cost and high performance.

In, an adaptive low-cost and high-quality image scalar was proposed, in which a sharpening spatial and a clamp convolution filters were used as prefilters to reduce the blur and aliasing effects produced by the bilinear interpolation. Although the chip area and memory requirement were efficiently decreased by hardware sharing and filter combining technique, it still demanded three multipliers, fifty adders, and a four-line-buffer memory. Thus, an adaptive edge-enhanced scaling algorithm that demands three multipliers, nineteen adders, and only a one-line-buffer memory is proposed in this paper.

II. BILINEAR INTERPOLATION, SHARPENING SPATIAL FILTER AND EDGE DETECTING TECHNIQUE

This section gives the brief discussion of bilinear interpolation, sharpening spatial filter and edge detecting technique.

A.Bilinear Interpolation

Bilinear Interpolation determines the grey level value from the weighted average of the four closest pixels to the specified input coordinates, and assigns that value to the output coordinates. It is an extension of linear interpolation for interpolating functions of two pixels. The key idea is to perform linear interpolation first in one direction, and then again in the other direction as shown in fig.1.

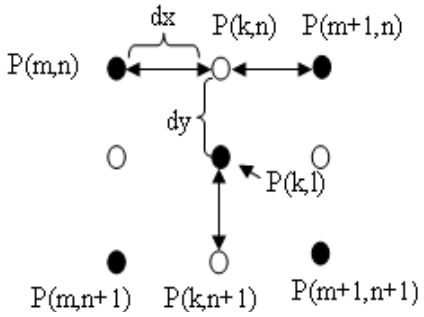

Fig. 1 Bilinear interpolation

Where dx is the scale parameter in horizontal direction and dy is the scale parameter in vertical direction. The M and N are the width and height of the original image.

B.Sharpening Spatial Filter

The sharpening spatial filter is a kind of high pass filter can be used to not only enhance the edges and details of the objects but also remove the low-frequency noise for image processing. It is defined by a kernel designed to increase the intensity of the center pixel relative to neighboring pixels. A convolution filter is a kind of linear filter, which modifies or enhances images by linear combination (for example addition and multiplication).
An example of the weights of five-pixel convolution function can be represented as

$$P'_{(m,n)} = [S \cdot P_{(m,n)} - P_{(m+1,n)} - P_{(m,n-1)} - P_{(m,n+1)} - P_{(m-1,n)}] \quad (1)$$

where S is a sharp parameter that adjusts the degree of sharpening effect.

Furthermore, to reduce the complexity and memory requirement of the cross-model convolution kernel, the sharpening spatial and clamp filters are realized by T model and inversed T-model convolution kernels as shown in fig. 2

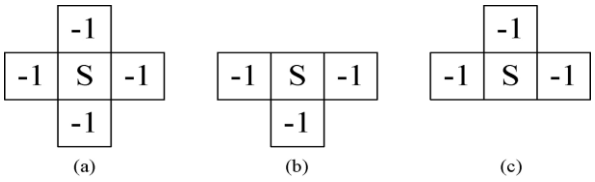

Fig. 2 Weights the of convolution functions. (a) Four neighboring pixels around the center pixel. (b) Center pixel relative to left, right and below neighboring pixels. (c) Center pixel relative to top, left and right neighboring pixels.

Although this convolution function only relates to the four neighboring pixels, it still needs at least a two-line-buffer memory to store the values of two-row pixels in the source image. In order to reduce the memory requirement of the sharp spatial convolution filter, two simplified convolution functions are created.

C. Edge Detecting Technique

An Edge is defined by the boundary with which it separates the higher intensity of the image with that of the lower intensities. An Edge can also be used as a filter in image processing. Edges characterize boundaries and are therefore a problem of fundamental importance in image processing. Edges typically occur on the boundary between two different regions in an image. Edge detection allows user to observe those features of an image where there is a more or less abrupt change in gray level or texture indicating the end of one region in the image and the beginning of another. A linear space-variant sigmoidal edge detecting model is selected as the edge detecting algorithm because of its low complexity and memory requirement. The asymmetry of the local characteristics neighbouring around P(k) can be evaluated by

$$D = |P(m+1) - P(m-1)|P(m+2) - P(m)| \quad (2)$$

where D is an asymmetry parameter defined to evaluate the asymmetry of the data in the neighborhood of P(k).By using sigmoidal functions, the asymmetry of the edges can be summarized as follows.
1) D=0, edges are symmetry on the left side and right side.
2) D>0, edges are more homogeneous on the right side.
3) D<0, edges are more homogeneous on the left side.

The sigmoidal function provides an efficient methodology to obtain the symmetry of edges by the nearest four neighboring pixels with simple operations. To avoid the edge features being lost after bilinear interpolation, the two pixels on the right-hand side are adaptively enhanced edge features by the sharpening spatial prefilter. The sigmodial edge detection is also efficient for implementing in VLSI.

III VLSI ARCHITECTURE

The VLSI architecture shown in fig. 3 consists of 5 blocks register bank, edge detector, sharp filter, bilinear interpolation as shown in figure

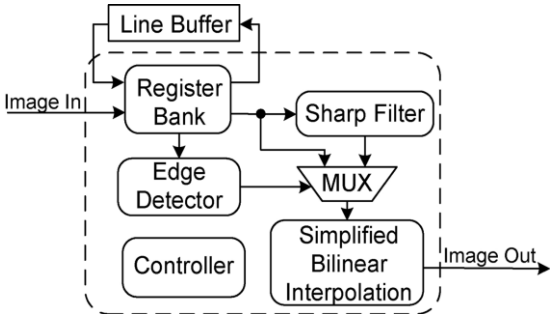

Fig. 3 VLSI architecture

A. Register Bank And Controller

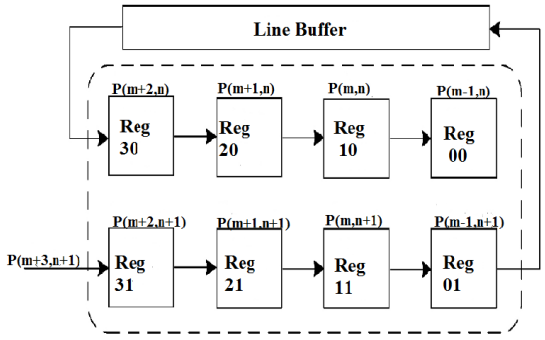

Fig . 4 Register bank

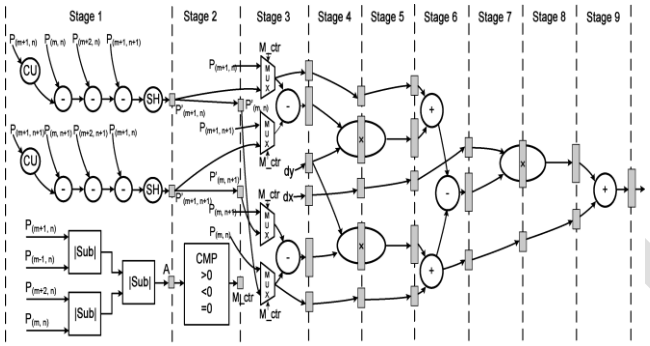

Fig . 5 Pipeline architecture

The fig. 4 shows the architecture of register bank.When the command is issued from the controller a new value P(m+3,n+1) is read into the Reg 31 and the stored contents will be shifted right. The value of Reg 01 is written into the line buffer and the value in leftest location in read into Reg 30. From this register bank design, the input pixels for edge detector and sharpening spatial filer can be obtained from the values of these eight registers. The line buffer memory can be reduced from two to one successfully. The controller produces the control signals for the register bank ,it is realised by using the finite state machine(FSM). It also produce control signals to control the pipelining and timing schedule of the bilinear
interpolation.

B.Pipeline Architecture

The figure shows the nine stage pipeline architecture of the adaptive edge-enhanced image scalar. By using the pipeline architecture the performance improves by shortening the delay path. the upper part of the first two-stage shows the pipelined architecture of the sharp filter. The lower part of the first two-stage in Fig. 9 shows the architecture of the edge detector that consists of three |Sub|units, one comparator (CMP), and two registers. Stages 3 to 9 in Fig. 5 show the architecture of the simplified bilinear interpolation circuit, in which the seven-stage pipelined architecture and the two-stage pipelined multipliers are used to shorten the delay path.

C. Reconfigurable Calculation Unit

In the sharpening spatial filter by using the hardware sharing technique the two of four filtering results and can be replaced by the registered temp values. It successfully reduces two filtering operations by only adding two registers. The pipeline architecture of sharp filter consists of two calculation units (CU), two shifters (SH), six subtracters (implemented by adders), and four registers.

The calculation unit shown in fig .6 consists of four shifters one multiplexer and two error tolerant adders. The multiplexer result is produced by selecting one value from the results of 2-bit, 3-bit, and 4-bit shifters, which calculates 4, 8, and 16 times of the input P. The adder produces three times of the value of the input P by adding the result of the 1-bit shifter and the value of the input P. After adding the results of the multiplexer and the adder, the outcome of the calculating unit can be calculated as 7, 11, or 19 times the value of the input P. Users can set the selecting control signals for the multiplexer (MUX) in each calculating unit according to the characteristics of images. By this design, the hardware cost of the calculation unit can be efficiently reduced.

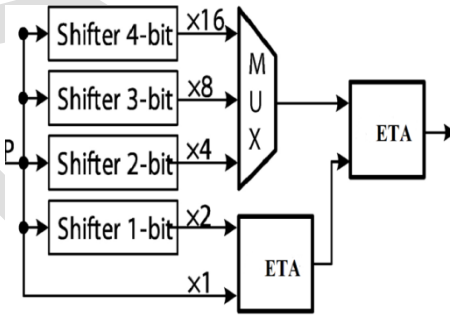

Fig . 6 Calculation unit

D. Error Tolerant Adder

The adders in the RCU design, uses the error tolerant adder method, which separates the pixel value into: an accurate part that includes several higher order bits and the inaccurate part that is made up of the remaining lower order bits. The addition process starts from the middle (joining point of the two parts) toward the two opposite directions simultaneously. The addition of the higher order bits (accurate part) of the input operands is performed from right to left (LSB to MSB) and normal addition method is applied. This is to preserve its correctness since the higher order bits play a more important role than the lower order bits. The lower order bits of the input operands (inaccurate part) require a special addition mechanism. No carry signal will be generated or taken in at any bit position to eliminate the carry propagation path. To minimize the overall error due to the elimination of the carry chain, a special strategy is adapted, and can be described as follow: 1) check every bit position from left to right (MSB to LSB); 2) if both input bits are "0" or different, normal one-bit addition is performed and the operation proceeds to next bit position; 3) if both input bits are "1," the checking process stopped and from this bit onward, all sum bits to the right are set to "1."

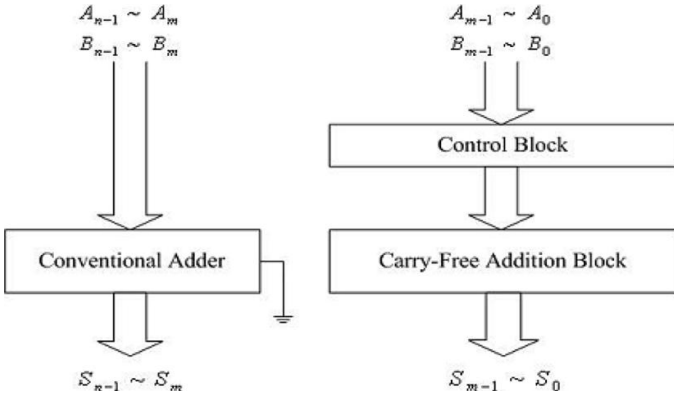

Fig . 7 Architecture of ETA

Fig . 7 shows the architecture of ETA. The accurate part consists of a conventional adder which performs normal addition. The inaccurate part consists of two parts namely the control block and carry free addition block. Bit B0 serves as the control bit for both accurate and inaccurate parts. If B0 is 1 adder performs the normal addition and if B0 is zero, the line from supply to ground is cut off and hence reducing the power dissipation.

A conventional full adder is used in the accurate part. It is the inaccurate part that decides the speed, accuracy and the power consumption of the adder. The carry free addition block has 4 modified XOR gates to give sum bits for LSBs. The inaccurate part has a CTL which controls the output of carry free addition block. When both or one of the inputs is zero, CTL is off and as soon as both the inputs are '1' it goes to logic '1'. Hence after this at least one of the inputs is always '1' so we get '1' as the output for any input that comes after this.

IV. RESULTS AND DISCUSSION

The VHDL coding is used to implement the existing and proposed methods. XILINIX gives the power analysis report by setting the family as spartan2E, device as XC2S600E and speed as -6.The total power consumption for existing system is 283mW, but the total power consumption for proposed system is 183mW.By looking at this results the power reduces by 100mW by using the error tolerant adder in the CU instead of the conventional adder.

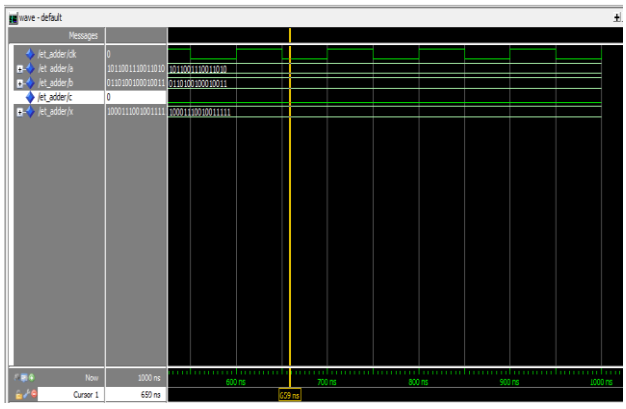

Fig . 8 Output of ETA

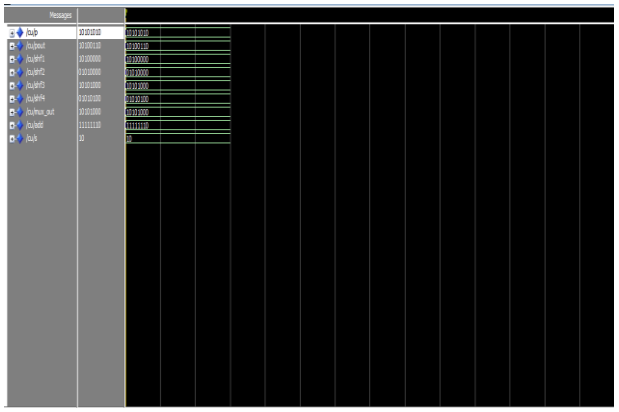

Fig . 9 Output of calculation unit

Power summary:	I(mA)	P(mW)
Total estimated power consumption:		183
Vccint 1.80V:	98	177
Vcco33 3.30V:	2	7
Clocks:	75	135
Inputs:	8	15
Logic:	0	0
Outputs:		
Vcco33	0	0
Signals:	0	0

Fig. 10.Power report of proposed method

Using MATLAB code the original image can bilinearly interpolated and the images can be compared as shown in Fig.9. By comparing to previous normal adder the proposed ETA consumes less power because of reduction of unwanted glitches.

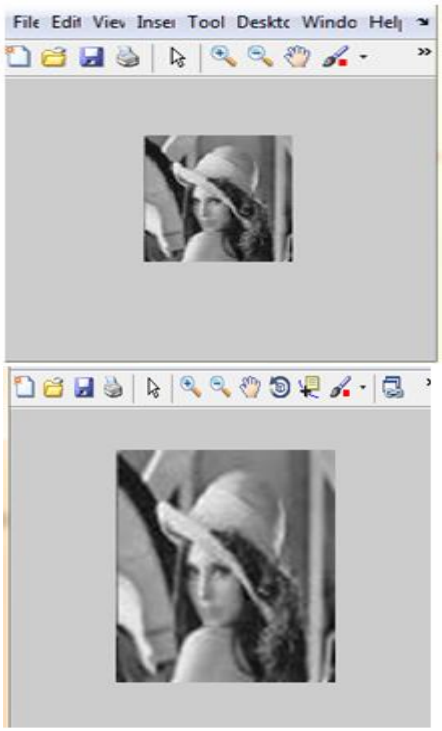

Fig.11.Comparision of images

The below table figures out the power consumption of adaptive scalar, image scaling processor, the existing method and proposed method. The table also reveals the gate count of various methods. Compared to various

methods the proposed method consumes less power and occupies less area and achieves great improvement in speed.

Methods	Area (Gate Count)	Power (mw)
Adaptive scalar	18,717	384
Image Scaling Processor	3745	316
This work(Existing)	2872	283
This work(proposed)-ETA	1840	183

Table1-simulation results of ETA versus image scaling methods

V. CONCLUSION

In this paper, a novel adaptive edge-enhanced technique was presented to develop a low-cost, low-power, low memory requirement, high quality, and high performance VLSI scalar circuit for real-time image zooming applications. The concept of error tolerance is introduced in VLSI design. A novel type of adder, the error-tolerant adder, which trades certain amount of accuracy for significant power saving and performance improvement. Extensive comparisons with conventional digital adders showed that the proposed ETA outperformed the conventional adders in both power consumption and speed performance. The potential applications of the ETA fall mainly in areas where there is no strict requirement on accuracy or where super low power consumption and high-speed performance are more important than accuracy.

REFERENCES

[1] V. Caselles, J. M. Morel, and C. Sbert, "An axiomatic approach to image interpolation," IEEE Trans. Image Process., vol. 7, no. 3, pp.376–386, Mar. 1998.

[2] E. Meijering, K. J. Zuiderveld, and M. A. Viergever, "Image reconstruction by convolution with symmetrical piecewise nth order polynomial kernels," IEEE Trans Image Process., vol. 8, no. 2, pp. 192–201,Feb. 1999.

[3] K. Jensen and D. Anastassiou, "Subpixel edge localization and the interpolation of still images," IEEE Trans. Image Process., vol. 4, no. 3, pp. 285–295, Mar. 1995.

[4] W. Y. V. Leung, P. J. Bones and R. G. Lane, "Statistical interpolation of sampled images," Opt. Eng., vol. 40, no. 4, pp. 547–553, Apr. 2001.

[5] C. H. Kim, S. M. Seong, J. A. Lee, and L. S. Kim, "Winscale : an image scaling algorithm using an area pixel model", IEEE Trans. Circuits System Video Technol., vol. 13, no. 6, pp. 549–553, Jun. 2003.

[6] Andreadis and A. Amanatiadis, "Digital image scaling," in Proc. IEEE Instrum. Meas. Technol. Conf., May 2005., pp. 2028–2032.

[7] S. L. Chen, H. Y. Huang, and C. H. Luo, "A low-cost high-quality adaptive scalar for real-time multimedia applications," IEEE Trans. Circuits Syst. Video Technol., vol. 21, no. 11 , pp. 1600–1611, Nov.2011.

[8] Shih-Lun Chen, "VLSI Implementation of an Adaptive Edge-Enhanced Image Scalar for Real-Time Multimedia Applications", IEEE Trans. Circuits and Systems for Video Technology., vol. 23, no. 9, Sep 2013.

[9] Ning Zhu, Wang Ling Goh, Weija Zhang, Kiat Seng Yeo, and Zhi Hui Kong, "Design of Low-Power High-Speed Truncation-Error-Tolerant Adder and Its Application in Digital Signal Processing", IEEE Transactions On Very Large Scale Integration (VLSI) Systems, vol. 18, no. 8, Aug 2010.