International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 13 (2018) pp. 11025-11029

Power Efficient Architecture for Image Scaling Applications

Dr. Narendra.C.P

*Associate Professo**r***
Department of Electronic and Communication
Bangalore Institute of Technology, Bangalore, India.

Chaitra G.

Department of Electronic and Communication
Bangalore Institute of Technology, Bangalore, India.

H.S.Pooja

Department of Electronic and Communication
Bangalore Institute of Technology, Bangalore, India.

Akash Kumar

Department of Electronic and Communication
Bangalore Institute of Technology, Bangalore, India.

Akash Deo

Department of Electronic and Communication
Bangalore Institute of Technology, Bangalore, India.

Abstract

One among the most widely used techniques for fitting images to the displays of the portable devices is digital image processing. Existing algorithms for image scaling consumes more power and area, thus can be rendered inefficient in the growing era of digital electronics. An algorithm of low complexity for image scaling is implemented in this work. In the implemented algorithm, the output pixel is calculated by either bilinear interpolation or replication. Energy efficiency of the design is achieved by introducing a module for edge catching to determine the computation method. Further manipulations are done algebraically, which results in pipelined architecture. The architecture proposed is implemented as a Verilog HDL code and simulated in ModelSim to verify the correctness. Further, the architecture was implemented in Xilinx ISE for FPGA synthesis and in Cadence for ASIC synthesis. The Cadence synthesis (when 45nm technology is used) yields approximately 20.29% reduction in power when compared to existing edge technologies.

INTRODUCTION

Since several decades, digital image processing is considered one among the major research areas. Among several applications, image processing is widely used in the areas of medicine [1], computer graphics [2], surveillance and consumer electronics [3] (e.g. mobile phones, High Definition TVs, etc). The increase of such portable electronic devices increases the demand for low power algorithms for image scaling.

The algorithms developed for scaling can be bifurcated into polynomial and non-polynomial based. This differentiation depends on the approach used. Bicubic [4], bilinear [5] and nearest neighbor [6] algorithms are polynomial based scaling algorithms. Orientation-adaptive interpolation, curvature interpolation [7] and adaptive 2-D autoregressive are non-polynomial based scaling algorithms.

Amongst all these methods, nearest-neighbor method is the simplest one since the pixels are duplicated based on its distance to the target pixel. Though computationally simple, blocking and aliasing effects are observed in this method. Bilinear interpolation, commonly used algorithm for scaling, computes the target pixel by considering weighted average of the nearby pixels. Bicubic interpolation is known to increase quality by compromising the simplicity required for scaling algorithms. The target pixel is obtained using 4x4 neighbouring pixels. The high frequency response degrades in both of these methods leading towards a blurred output image. Many computationally complex polynomial based algorithms exist which delivers better quality but at the cost of increased hardware and power. Hence, architectures which are efficient in terms of power are required.

Architectures for implementation which are less complex are introduced in recent times. Winscale algorithm by Kim et al. [8] uses domain filtering and area pixel concept for calculation of interpolated pixel. Lin et al proposed an efficient architecture for this algorithm [9]. This method, although proven to preserve edge characteristics, require more computations than bilinear method. Since it also needs memory for storing pixels, the cost of implementation increases. Further, efficient architecture of bicubic interpolation [10] was proposed. This needs large number of computations for calculating intermediate pixels on vertical and horizontal directions, which also increased memory requirement. A cost effective design, adaptive in nature, for image interpolation of real time images which use a four line buffers and combined filter was presented by Chen en tal [11] Though good results were obtained, area overhead was considerably increased because of use of four line buffers. Four line buffers used were optimized to one line buffer in [12]. Low memory requirement and better quality was achieved in edge enhanced algorithm as in [13]. Though most

of these techniques target complexity and area, power efficiency is a prime concern.

In this work, efficient architecture for image scaling processor for multimedia applications is proposed in which target pixels are obtained by bilinear interpolation near the edges and by replication for non-edge pixels.

METHODOLOGY

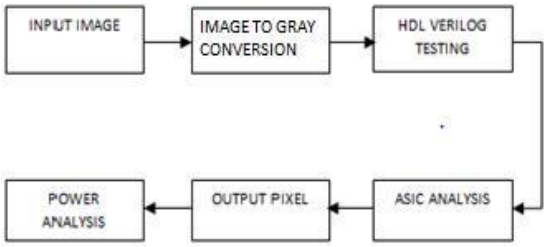

Figure 1. Proposed Methodology

The input image is applied to MATLAB, the resulting pixel is scaled and the processed pixel is undergone and analysis to determine the power utilized.

ARCHITECTURE

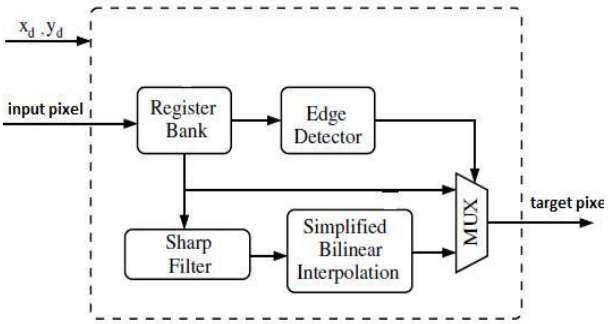

Figure 2. Image Scaling Processor

The block diagram defines the processor designed for image scaling. It includes register bank, edge detector, sharp filter, simplified bilinear block and MUX.

i. Register Buffer Bank

The pixel values of input image stored in register banks in the form of rows so that they can be used for both sharpening filter and edge catching block operations.

ii. Edge Detector Block

This block denotes the presence of edge in the image. For the implemented architecture, the edge detection takes place by considering four pixels at th, each time the value of a1 and a2 is calculated as follows

$$a1 = r(i+1,j+1) - r(i+1,j) \quad (1)$$

$$a2 = r(i,j+1) - r(i,j) \quad (2)$$

The error is detected in either of the cases a1>a2 or a2<a1(for heterogenous). This is implemented in the proposed design using exor gate.

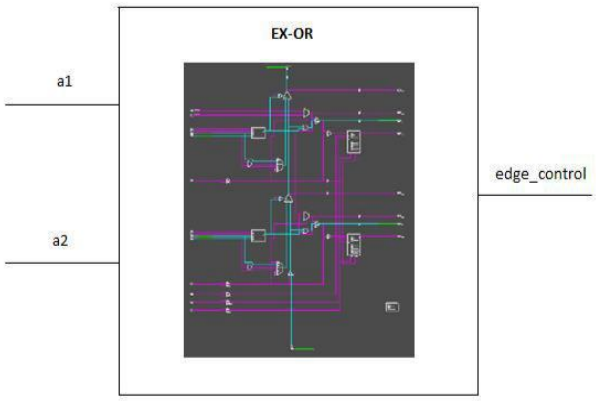

Figure 3. Comparator

The uses of this gate instead of comparator reduces the complexity of circuit and power by 20% approximately and as a result increase in slight delay due to increased number of iterations. This also reduces the area occupied by the circuit.

iii. Sharp Filter

The image has to be sharpened before undergoing bilinear interpolation in order to prevent blurring of image. This is carried out using the following equations

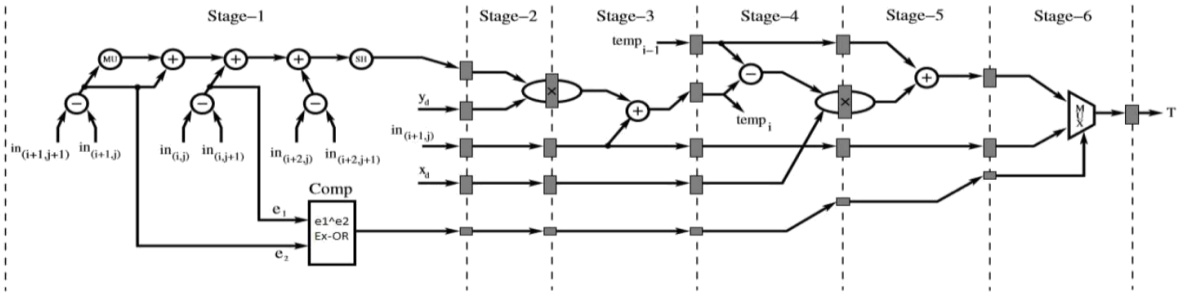

Figure 4. Proposed Pipelined Architecture[14]

$$rs(i+1,j+1) = [S\ r(i+1,j+1) - r(i+2,j+1) - r(i+1,j) - r(i,j+1)]/S\text{-}3 \quad (3)$$

$$rs(i+1,j) = [S\ r(i+1,j) - r(i+2,j) - r(i+1,j+1) - r(i,j)]/S\text{-}3 \quad (4)$$

$$rs(i,j+1) = [Sr(i,j+1) - r(i+1,j+1) - r(i,j) - r(i1,j+1)]/S\text{-}3 \quad (5)$$

$$rs(i,j) = [Sr(i,j) - r(i+1,j) - r(i,j+1) - r(i,-1,j)]/S\text{-}3 \quad (4)$$

The S is the scaling factor which is user defined.

iv. Simplified Bilinear Interpolation

The sharpened pixel is undergone bilinear interpolation by using the formula

$$T(p,q) = \{[rs(i+1,j) + yd \times (rs(i+1,j+1) - rs(i+1,j))] - [rs(i,j) + yd \times (rs(i,j+1) - rs(i,j))]\} \times xd + [rs(i,j) + yd \times (inS(i,j+1) - rs(i,j))]\} \quad (7)$$

The hardware cost, area is reduced by using the input pixel r(i+1,j) directly at stage 2. This reduces the number of multiplication unit by 1and subtractor by 4and shifter by 1.

The Verilog HDL code for the architecture proposed was synthesized in the Xilinx ISE tool for the implementation in FPGA. The architecture implemented in this work requires only 3 multipliers and 11 adders which shows reduction in the number of adders and multipliers when compared to the numbers used in many other methods. The number of adders and multipliers used for existing architectures are compared in the table I below. The proposed design is implemented using Xilinx ISE Spartan-3 XC3S504CP132 FPGA core. The synthesis results shows that the proposed design has a frequency of 143.554MHz and maximum period of 6.966ns.

Table I: Comparison of the resources used in architectures

Units	BL [5]	BC [6]	Adaptive [3]	Mod Ada [11]	Edge[13]	Edge Ada [14]	Implemented
Adders	7	32	3	4	3	2	3
Multipliers	7	33	50	36	19	10	11

The synthesis of Verilog HDL code is done on Cadence for ASIC implementation. The library used is fast.lib. The technology used is 45nm technology. It is observe that when timing is constrained to 1000ps, the power increases drastically. This is depicted in table II.

Table II. Comparison of power for constrained and unconstrained timing

Type of timing analysis	Power (in nW)
Unconstrained	407859.856
Constrained to 1000ps	3029357.448

The proposed architecture uses carry save adder which results in decrease in total power consumption.

RESULTS

The architecture was implemented in Verilog HDL. Simulation was done in ModelSim for both the cases. Figure 5.1 depicts the results for edge case (bilinear interpolation case) and Figure 5.2 depicts the results for non-edge case (replication case). Muxout is the output and pixel1, pixel2 are input pixel values.

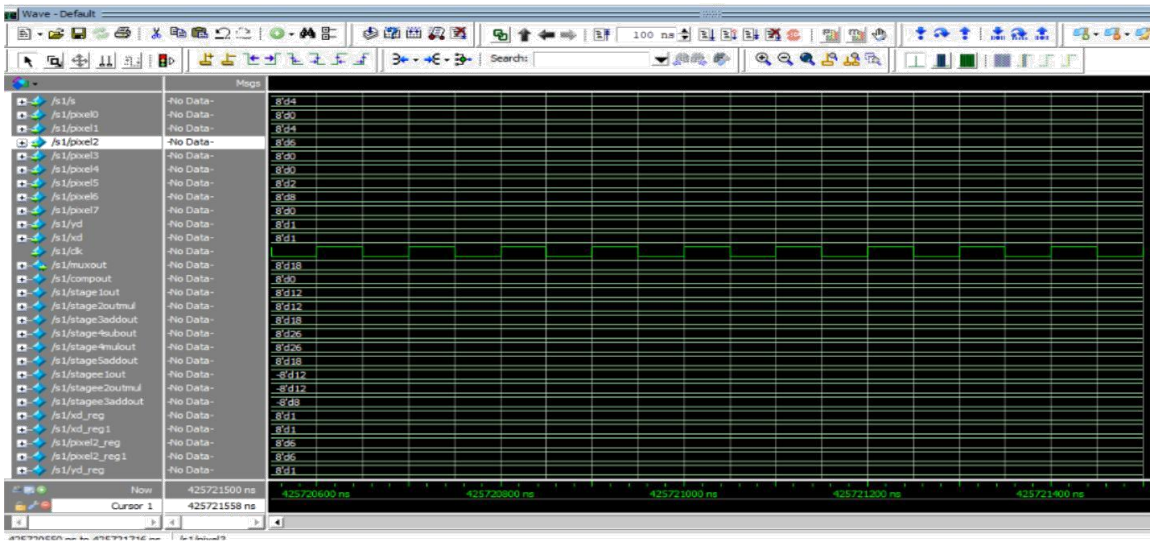

Figure 5.1 Waveform of simulation for first case (bilinear interpolation).

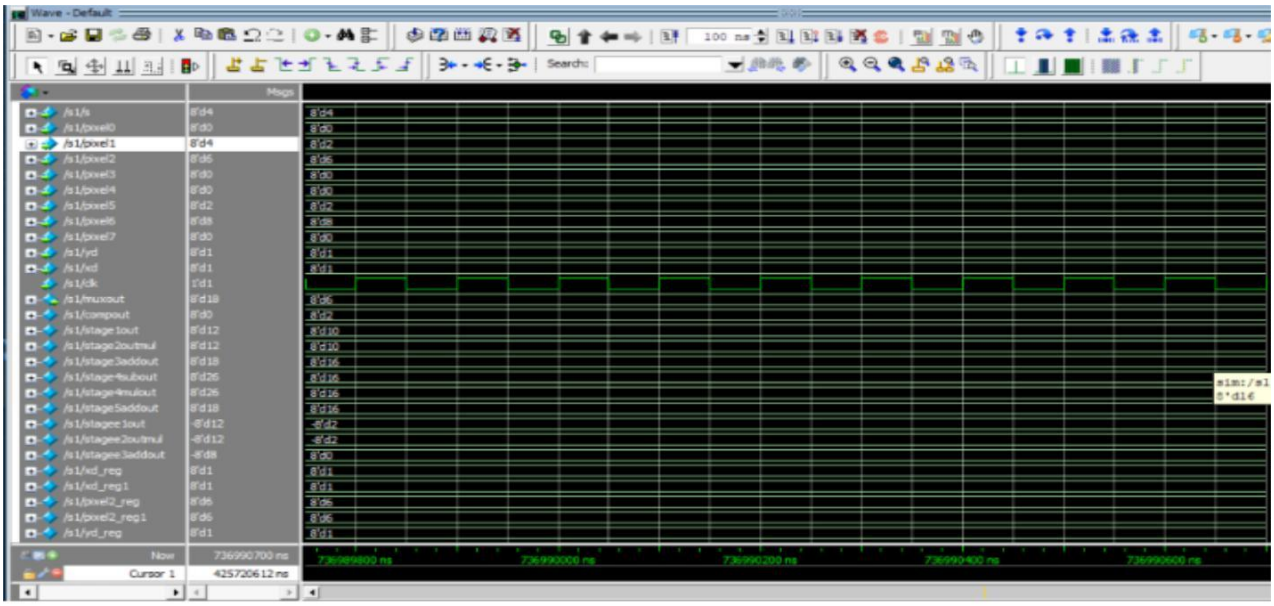

Figure 5.2. Waveform of simulation for the second case (replication of the pixel)

This work proposes an architecture with reduced power when compared to edge technology [13]. This is shown in table III. It is evident from the table that there is a decrease of power by 20.29% in the architecture implemented when compared to [14].

Table III: Design metrics comparison

	Edge[14]	Implemented
Power	2.01nW	.4078nW

CONCLUSION AND FUTURE WORK

This paper gives a power efficient architecture for image scaling by using edge detection. The proposed architecture results in 20% reduction in power. The architecture can be used in low power consumption applications.

The delay can be further reduced by using different type of multipliers, adders and effective use of the 4 DSPA blocks in FPGA.

REFERNCES

[1] D.-H. Trinh, M. Luong, F. Dibos, J.-M. Rocchisani, C.-D. Pham, and T. Nguyen, "Novel example-based method for super-resolution and denoising of medical images," Image Processing, IEEE Transactions on, vol. 23, no. 4, pp. 1882– 1895, April 2014.

[2] W. Freeman, T. Jones, and E. Pasztor, "Example-basedsuper-resolution," Computer Graphics and Applications, IEEE, vol. 22, no. 2, pp. 56–65, Mar 2002.

[3] A. Amanatiadis and I. Andreadis, "An integrated architecture for adaptive image stabilization in zooming operation," Consumer Electronics, IEEE Transactions on, vol. 54, no. 2, pp. 600–608, May 2008.

[4] R.Keys, "Cubic convolution interpolation for digital image processing," Acoustics, Speech and Signal Processing, IEEE Transactions on, vol. 29, no. 6, pp. 1153–1160, Dec 1981.

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

[6] V. Caselles, J.-M. Morel, and C. Sbert, "An axiomatic approach to image interpolation," in Image Processing, 1997. Proceedings., International Conference on, vol. 3, Oct 1997, pp. 376–379.

[7] H. Kim, Y. Cha, and S. Kim, "Curvature interpolation method for image zooming," Image Processing, IEEE Transactions on, vol. 20, no. 7, pp. 1895–1903, July 2011.

[8] C.-H. Kim, S.-M. Seong, J.-A. Lee, and L.-S. Kim, "Winscale: an image scaling algorithm using an area pixel model," Circuits and Systems for Video Technology, IEEE Transactions on, vol. 13, no. 6, pp. 549–553, June 2003.

[9] C. chi Lin, Z. chuan Wu, W. kai Tsai, M. hwa Sheu, and H.-K. Chiang, "The VLSI design of winscale for digital image scaling," in Intelligent Information Hiding and Multimedia Signal Processing, 2007. IIHMSP 2007. Third International Conference on, vol. 2, Nov 2007, pp. 511–514.

[10] C. chi Lin, M. hwa Sheu, H.-K. Chiang, C. Liaw, and Z. chuan Wu, "The efficient VLSI design of bi-cubic convolution interpolation for digital image processing," in Circuits and Systems, 2008. ISCAS 2008. IEEE International Symposium on, May 2008, pp. 480–483.

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

[12] S. L. Chen, "VLSI implementation of a low-cost high-quality image scaling processor," Circuits and Systems II: Express Briefs, IEEE Transactions on, vol. 60, no. 1, pp. 31–35, Jan 2013.

[13] S. L. Chen, "VLSI implementation of an adaptive edge-enhanced image scalar for real-time multimedia applications," Circuits and Systems for Video Technology, IEEE Transactions on, vol. 23, no. 9, pp. 1510–1522, Sept 2013.

[14] Bharat Garg, V N S K Chaitanya Goteti and G K Sharma, "A Low-Cost Energy Efficient Image Scaling Processor for Multimedia Applications".