Prediction of Approximate Multiplier for 16-Bit DICOM Image Contrast Scaling Using Classical Machine Learning Approach

maria dominic (✉ mariadom@srmist.edu.in)

SRMIST: SRM Institute of Science and Technology

DEEPA T

SRMIST: SRM Institute of Science and Technology

Research Article

Keywords: Image processing, NED, MED, Multiplier, Approximate Compressor

Posted Date: May 7th, 2021

DOI: <https://doi.org/10.21203/rs.3.rs-444275/v1>

License: This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Prediction of Approximate Multiplier for 16-Bit DICOM Image Contrast Scaling Using Classical Machine Learning Approach

M.Maria Dominic Savio,T.Deepa

Received: date / Accepted: date

Abstract The role of approximate arithmetic are involved when the processors are used for multimedia signal processing application. The impact of multiplier is very important in many processes done by these processors. The compressors are the core architecture for reduction stage if the multiplier width is increased. Later approximations are done in the compressor to limited error without affecting the signal standard. The design of scalable-split compressor is designed in this work and a counter matching method has been developed for approximation. The design of 32x32 and 16x16 multiplier with these new compressors are synthesised in 45nm Synopsis Design Compiler and shows an improvement of 25 % of Chip area and 27% power. The split-scalable architecture attempts to reduce the delay with trade-off in area and power. Mean Error Distance (MED) and Normalized Error Distance (NED) are the parameters that ensure the quality of any approximate arithmetic based design. 16-bit medical images are processed with both existing and proposed multipliers then the Peak Signal to Noise Ratio (PSNR) is compared. Finally with several input nature and targeted PSNR the best system is identified using classical machine learning model.

Keywords Image processing · NED · MED · Multiplier · Approximate Compressor

1 Introduction

The digital processing system (DPS) plays a significant role in today's real-time application [1]. The design components of DPS are adder, subtractor, multiplier, shifter, and comparator to perform the necessary operations. The

Maria Dominic Savio ,T.Deepa
SRM Institute of Science and technology,Chennai, 603203, Tamilnadu, India.
Tel.:9994076650
E-mail: mariadom@srmist.edu.in

design of the multiplier is the most complex task in the construction of DPS. The multipliers consume a higher amount of energy and more delay among all design components [2]. The basic operations like arithmetic computation, comparison, multiplication of signals are performed by DPS's [3]. These systems are embedded with the arithmetic blocks, which provide high accuracy and reliability [4]. When these arithmetic blocks are used in an image processing the approximate arithmetic is carried out in order to reduce the design complexity with minimal error without degradation in the application's performance, [5]. The method of approximate computing is carried out by connecting some output function as the direct input without involving the circuit design. Approximate subtractors are developed in [6] with Boolean solvation using Karnaugh map (K-map), and the approximate divider is designed using the same subtractor for subtract two image to achieve background subtraction. To reduce power, area and increase the speed several approximate comparators in [7]-[9] are designed. The salt and pepper noise are removed with these comparator with no degradation in image quality. The design of compressors play the best alternative for full-adder in Wallace and Dadda multiplier for dot product reduction [10]. Compressors are usually developed with full-adders, an alternative in [11] compressor 4:2, 5:3 are designed using XOR-MUX.

Various multiplier architectures for image processing applications have been done with imprecise compressors proposed in [12] – [14] but the overall errors are limited. The multipliers are constructed with three steps process as follows: (i) dot product generation (ii) reduction of dot product using suitable tree (iii) addition of dot product using adders [15]. Among the process, the second stage is more complicated because it consumes more energy, and produces the delay. The Wallace and Dadda tree architecture is not an efficient method for dot product reduction if the full-adder is used [16]. In the reduction stage, the replacement of full adders is done with a compressor to reduce the design complexity [17]. The 4:2 compressors are the perfect fit for all kind of multipliers with lower width. In the field of medical imaging the requirement of 16-bit pixel images are used to improve the accuracy it is different from the nature of 8-bit pixels [18]. When designing 16x16 or any other higher width multipliers using 4:2 compressor the reduction stage gets increased. Hence, in [19], 16x16 multiplier is constructed with 15:4 compressor.

In multiplier, the approximate compressor acts as a key role for the reduction of area power and delay [20]. For a lower order compressor, the approximation is achieved through truth table and K-map. In truth table matching of input to output is identified, then the highly correlated values are connected straightly. In K-map the device hardware was reduced by removing the essential prime implicant [21]. The image quality and device hardware are equally concentrated in all the approximate computing methods. The image quality is measured through peak signal to noise ratio (PSNR). The variation between two digital images is identified using PSNR [22]. The PSNR value of 30dB is enough for most of the image processing applications [23]-[28]. The overall multiplier quality is measured through error distance (ED), which represent difference between the accurate product and the approximate product [29].

The design of several 9:4, 8:4, 8:2, 7:2 compressors are proposed in [30]-[33]. They have been designed with sub-component of three or more small compressor. In [18] the ED of 15:4 compressor is calculated through 5:3 compressor, which is the internal module of 15:4 compressor.

The novel counter based comparison method between inputs and outputs has been addressed in this paper. The proposed compressors are efficient in terms of area, power, and delay, NED and MED. The data-set is created for various standard medical images from different database with the values of input parameter like nature of image, size, contrast ratio, and output PSNR. Artificial Neural Network(ANN) and Logistic Regression(LR) that improved in [34] is used to predict the best approximate multiplier depend on the metrics matched with the newly applied image. The upcoming sections of the paper narrates. Section II explains split-scalable compressor design. The design of impreciseness using a counter based comparison method is discussed in section III. Higher width multiplier architecture discussed in IV. The system performance analysis is delivered in section V. Multiplications of medical images and standard test images are done with the proposed multiplier is given in section VI. Machine learning prediction is discussed in section VII. At the last conclusion is presented in section VIII.

2 DESIGN OF SCALABLE-SPLIT COMPRESSOR

The design of several 15:4, 9:4 and 8:4, 8:2, 7:2 compressors are proposed by several researchers. These designs are done with sub-component of three or more small compressors. These small compressors are developed with XOR-MUX instead of full-adder proposed by many authors as shown in Fig.1. The approximation is the tedious process when higher compressor are used.

Fig. 1 (a) Full Adder (b) Xor mux

The proposed scalable compressors are designed with a stream line approach of parallel inputs to outputs through XOR-MUX architecture as shown in Fig.2. The term scalable means that the proposed compressor can be extended to any higher-order level. In Fig.2 the different levels of the compressor are differentiated with a dotted square box, with the same approach N:2 compressor can be constructed with $(N-3)$ $C_{in}'s$ and $C_{out}'s$ where N number of inputs and the two system outputs are sum and carry.

Designing of the higher-order compressor (8:2) using four 4:2 compressors was developed in [35]. For any compressor, XOR of all inputs computes the sum, and every C_{out} is calculated from the MUX output. The select line for MUX is computed from XOR of the first two input values; weather MUX inputs are first and third inputs. The sum for the 5:2 stage is given in equation (1), and carry based on XOR is given equation (2).

$$sum = x0 \oplus x1 \oplus x2 \oplus x3 \oplus x4 \oplus ci0 \oplus ci1 \quad (1)$$

$$carry = ((a \oplus b)c) + ((a \oplus b)a) \quad (2)$$

Fig. 2 Salable Split compressor

The proposed compressors do not consist of any lower-order compressor. As shown in Fig.2, each dotted box the final XOR output (X) that is passing to the next stage is the sum output, and the output collected from MUX ($X - 1$) is the carry for that stage obeying (N-3) $C_{out's}$, $C_{in's}$. This work has to target

Fig. 3 Critical path in compressor (a) longest path (b) Splited path

for 32x32 multiplication, so up to 32:2 compressor was designed. The concept of split compressor is introduced on scalable compressor as shown in Fig.3. The upper and lower split increases the design complexity of the tree stage in multiplier but reduce the critical path delay. The entire partial products are arrived after one gate delay associated with AND gate. As shown in Fig.3 the generation of sum takes the longest path and affects multiplier performance. So generation of two sum terms using split compressor will minimize the delay. Though upper and lower half operate in parallel to produces two sum term and half adder is used to reach the final sum term, so this structure reduces delay from T to $T/2 + 1$.

3 DESIGN OF COUNTER-MATCHING FOR APPROXIMATION

The approximate computing is the state of art method for reducing the energy, chip area and to increase the speed. The comparator and counter-based novel approximation techniques are presented in this paper. The circuit will check for 2^N samples, where N is the number of inputs. For example 5:2 compressor consists of 7 inputs a_0-a_4 , $C_{in0}-C_{in1}$ and having $2^7 = 128$ input combinations. The circuit consists of 4 outputs $C_{out0} - C_{out1}$, sum, carry. This work overcomes the approximation done in [19] by matching each inputs to every outputs for all possible input samples that can be applied to approximate any part of the circuit. The approximation finder blocks is described in Fig.4, which tells how many combination of the input's and output's are same. The Pseudo-code for approximation is shown algorithm-1, which demonstrates every input's correlation to every output. The proposed approximation finder circuit consists of $J = N \times M$ counter and comparator. All 1-bit comparator inputs are connected with compressor inputs and outputs. The 1st comparator is connected with a_0 and C_{out0} , 2nd comparator is connected with a_0 and C_{out1} , likewise J^{th} comparator connected with N and M . All J -counter outputs predict the count of matched input and output combination. With the counter output the various level of approximation will be performed to maintain the image quality.

Fig. 4 Approximation finder

Algorithm 1: Approximation

```

1 initialize
2 input of Compressor i=1 to N;
3 output of Compressor x=1 to M;
4 Counter  $J_{yz}$ ;
5 where  $y = 1$  to  $N \times M$  and  $z = 1$  to  $2^N$ ;
6 Number of Cycles  $K = 1$  to  $2^N$  ;
7 L1:for  $K=0$  to  $\leq 2^N$  &  $K++$ ;
8 do  $\rightarrow$  Set  $y=0$ ;  $i=0$ ;  $x=0$ ;
9 L2:for  $i=0$  to  $\leq N$  &  $i++$ ;
10 L3:for  $x=0$  to  $\leq M$  &  $x++$ ;
11 do  $\rightarrow$ 
12 if  $i = x$  then
13 set  $\rightarrow y++$ ;  $z++$ ; set  $\rightarrow y++$ ;  $z=z$ ;
14 end:L3
15 end:L2
16 end:L1
17 Collect all counter value  $J_y$  that gives mathed value  $J_z$ 

```

4 MULTIPLIER DESIGN

Most of the recent multiplier are constructed with compressor. In [12] discrete cosine transform (DCT) operation are done with approximate compressor. In this work many new designs of the approximate multiplier are developed. Design-1 32x32 multiplier with scalable compressor. Design-2 32x32 multiplier with split-scalable compressor, Design-3 32x32 multiplier with approximate scalable compressor approximating LSB. Design-4 32x32 multiplier with approximate scalable-split compressor in upper-split of LSB area, Design-5 32x32 multiplier with approximate scalable-split compressor in lower-split of LSB area, Design-6 32x32 multiplier with approximate entire scalable-split com-

pressor in LSB area, Design-7 32X32 multiplier using 15:4 compressor [19], Design-8 16x16 multiplier with exact scalable compressor, Design-9 16x16 multiplier with exact split-scalable compressor, Design-10 16x16 multiplier with the approximate scalable compressor in the LSB area, Design-11 16x16 multiplier with approximate split-scalable compressor in upper region of LSB, Design-12 16x16 multiplier with approximate split-scalable compressor in lower region of LSB, Design-13 16x16 multiplier with approximate on both upper and lower compressor in LSB, Design-14 16x16 multiplier with 15:4 compressors [19], Design-15 16x16 multiplier with 4:2 compressor [14].

4.1 Design-1 32x32 multiplier with scalable compressors

In [14] 8x8 multiplier is designed with a 4:2 compressor. The new multiplier is 32x32 width so the usage of any old type compressor increases the design complexity, power and area, so it is easy to design with scalable compressor as shown in Fig.5.

Fig. 5 32x32 multiplier with exact scalable compressor

4.2 Design-2 32x32 multiplier with the split-scalable compressor

To reduce delay of the proposed multiplier the concept of split compressor has been introduced. Here the maximum scaling length of one 17:2 compressor and one 16:2 compressor is used in the center part of the multiplier and either side is scaled down to fit with the bit length.

4.3 Design-3 32x32 multiplier with approximate scalable compressor in LSB area.

The exact compressor is replaced with an approximate compressor on the right most part of the multiplier from 3rd column to 32nd column.

4.4 Design-4 32x32 multiplier with approximate scalable-split compressor in upper-split of LSB area

The exact compressor is replaced with an approximate compressor on the right most part of the multiplier only in the upper half of the split compressors.

4.5 Design-5 32x32 multiplier with approximate scalable-split compressor in lower-split of LSB area

The exact compressor is replaced with an approximate compressor on the right most part of the multiplier only in the lower half of the split compressors.

4.6 Design-6 32x32 multiplier with approximate entire scalable-split compressor in LSB area

The exact compressor is replaced with an approximate compressor on the right most part of the multiplier on both upper and lower half of the split compressors.

4.7 Design-7 32X32 multiplier 15:4 compressor [19]

To verify the proposed approximate multiplier with previous work [19] and to compare the performance metrics the 32x32 multiplier is designed using an accurate 15:4 compressor developed in [19]. And the best approximate design from [19] is developed to compare the image quality of existing design with proposed designs done in this work.

4.8 Design-8 16x16 multiplier with scalable compressor

The scalable compressor are used to design the 16x16 multiplier is the same way as design-1 as shown in the Fig.5. The compressor with empty dot notifies the zero input just to compensate the two stage reduction architecture.

4.9 Design-9 16x16 multiplier with exact split-scalable compressor

To reduce delay of the scalable architecture the normal multiplier reduction stage are split with upper and lower compressor. The split has been implemented only the middle column of the multiplier as shown with dotted line in Fig.6.

Fig. 6 16X16 multiplier with split scalable compressor

4.10 Design-10 16x16 multiplier with the approximate scalable compressor in the LSB area

The exact compressor is replaced with an approximate compressor on the right most part of the multiplier from 3rd column to 16th column.

4.11 Design-11 16x16 multiplier with approximate split-scalable compressor in upper region of LSB

The exact compressor is replaced with an approximate compressor on the right most part of the multiplier only in the upper half of the split compressors.

4.12 Design-12 16x16 multiplier with approximate split-scalable compressor in lower region of LSB

The exact compressor is replaced with an approximate compressor on the right most part of the multiplier only in the lower half of the split compressors.

4.13 Design-13 16x16 multiplier with approximate on both upper and lower compressor in LSB

The exact compressor is replaced with an approximate compressor on the right most part of the multiplier on both upper and lower half of the split compressors.

4.14 Design-14 16x16 multiplier with 15:4 compressor with 5:3 as sub-component [19]

The 15:4 compressor is developed with 5:3 compressor and parallel adder as sub-components. In [19] the multipliers are constructed with 15:4 compressor, 5:3 compressor, full-adder, half-adder. The approximation method is only applied on 5:3 compressor. To compare existing model with straight forward approach of proposed multiplier and the performance are analyzed.

4.15 Design-15 16x16 multiplier with 4:2 compressor in [14]

In so many literature 4:2 compressor are used to develop the 8x8 multiplier. The 4:2 compressor as shown in Fig.1 are used to develop the 16x16 multiplier to compare the results with proposed multiplier.

5 RESULTS AND DISCUSSION

5.1 DELAY COMPARSION OF SCALABLE-SPLIT COMPRESSOR

The critical path delay has reduced to $T/2+1$ when the scalable-split compressors are used. The longest path in 16x16 multiplier is the 16th column where the Split compressors are used as shown in Fig.7. To compare delay in sum term the Xor path is traced between split and non split architecture and synthesised with 90 nm library in Cadance virtuso to compare the area, power and delay. Table-1 shows that delay of the split scalable compressor is reduced and more over the glitches also reduced.

Fig. 7 16:2 split compressor

Table 1 Comparison Split-scalable with Scalable compressor

Design	Area (gate count)	Power (uw)	Delay (ns)
Scalable Compressor	28	.428	34.4
Split-Scalable Compressor	28	.428	19.1

5.2 Approximation

The Verilog simulation result of 12:2 compressor approximations is shown in Fig .8. It is clearly shown that the counter values of $C_{in6}C_{out8}$, $C_{in7}carry$, and $C_{in8}carry$ are the maximum correlated for the total cycles. From the functional verification 75% of cycles many inputs are matched with outputs. TABLE-2 shows the highly correlated combination for the different higher-order compressor. From TABLE-2, so many approximations can be performed. But approximation over a least significant bit (LSB) reduces the error distance (ED) and normalized error distance (NED), so in all compressors, final Cin's are approximated as Carry to reduce the hardware.

The error value of new compressors are studied with the exact compressor. Equation (3-4) describe the NED and MED. .

$$MED = \frac{1}{2^{2N}} \sum_{i=1}^{2^{2N}} |ED_i| \quad (3)$$

$$NED = \frac{MED}{D} = \frac{1}{2^{2N}} \sum_{i=1}^{2^{2N}} \frac{|ED_i|}{D} \quad (4)$$

Where N is the input length of the multiplier, the ED_i is the difference between the exact and approximated multiplier of the i^{th} output vector. Where D is the maximum possible error is given by $(2^N - 1)^2$.TABLE-3 shows the NED and MED of the proposed and existing design

From the Fig.9, design-11 and design-12 shows the good performance and produces less error.

+ /e12/cin6cout8	1572864	1572850	1572851	1572852	1572853
+ /e12/cin7sum	1048576	1048569		1048570	
+ /e12/cin7carry	1572864	1572850	1572851	1572852	1572853
+ /e12/cin7cout0	1048576	1048568		1048569	
+ /e12/cin7cout1	1048576	1048565	1048566		1048567
+ /e12/cin7cout2	1048576	1048562	1048563	1048564	1048565
+ /e12/cin7cout3	1048576	1048562	1048563	1048564	1048565
+ /e12/cin7cout4	1048576	1048562	1048563	1048564	1048565
+ /e12/cin7cout5	1048576	1048562	1048563	1048564	1048565
+ /e12/cin7cout6	1048576	1048562	1048563	1048564	1048565
+ /e12/cin7cout7	1048576	1048562	1048563	1048564	1048565
+ /e12/cin7cout8	1048576	1048562	1048563	1048564	1048565
+ /e12/cin8sum	1048576	1048569		1048570	
+ /e12/cin8carry	1572864	1572850	1572851	1572852	1572853
Now	2097153 ps			2097140 ps	
Cursor 3	2097152 ps				

Fig. 8 Simulation Output for highly matched values

Table 2 Maximum correlated

input/output & design	correlated/ total cycle	input/output & design	correlated/ total cycle	input/output & design	correlated/ total cycle
a_0C_{out0}	6144/ & 8:2	a_0C_{out0}	393216/ & 11:2	a_0C_{out0}	25165824/ & 14:2
$C_{in4}carry$	6144/ & 8:2	$C_{in7}carry$	393216/ & 11:2	$C_{in10}carry$	25165824/ & 14:2
a_0C_{out0}	24576/ & 9:2	a_0C_{out0}	1572864/ & 12:2	a_0C_{out0}	100663296/ & 15:2
$C_{in5}carry$	24576/ & 9:2	$C_{in8}carry$	1572864/ & 12:2	$C_{in11}carry$	100663296/ & 15:2
a_0C_{out0}	98304/ & 10:2	a_0C_{out0}	6291456/ & 13:2	a_0C_{out0}	402653184/ & 16:2
$C_{in6}carry$	98304/ & 10:2	$C_{in9}carry$	6291456/ & 13:2	$C_{in12}carry$	402653184/ & 16:2

Table 3 Accurateness of 16x16 Multiplier

Design	NED	MED
Design-10	3.36×10^{-5}	6189×10^{11}
Design-11	2.11×10^{-5}	3908×10^{11}
Design-12	2.05×10^{-5}	3789×10^{11}
Design-13	3.53×10^{-5}	6553×10^{11}
Design-14[19]	5.43×10^{-5}	9989×10^{11}
Design-15[14]	4.17×10^{-5}	7488×10^{11}

5.3 Results of 16x16 bit multiplier and 32x32 multiplier

All the design of multipliers are functionally verified with Verilog HDL. The power, area, and delay are calculated in the Synopsys design compiler with a 45nm technology library. TABLE 4 and 5 shows the area, power, and de-

Fig. 9 Histogram for NED of 16X16 multiplier

lay values of 16x16 and 32x32 multipliers. From TABLE-4, design-3, design-6 shows good VLSI aspects compared to multipliers with existing compressors. Likewise, from TABLE-5 design-10, design-13, show improvement with exiting multiplier.

Table 4 Result Analysis of 32x32 multiplier

Design	Area(μm^2)	Power(mw)	Delay(ns)
D-1	2014	3.5	2.6
D-2	2351	4.3	1.8
D-3	1759	2.8	2.1
D-4	2112	3.4	1.47
D-5	2099	3.3	1.41
D-6	1849	2.1	1.12
D-7[19]	2049	3.41	2.9

Table 5 Result Analysis of 16x16 multiplier

Design	Area(μm^2)	Power(mw)	Delay(ns)
D-8	1036	1.71	1.4
D-9	1189	2.1	0.89
D-10	876	1.42	1.1
D-11	1040	1.8	0.72
D-12	1036	1.79	0.75
D-13	989	1.61	0.63
D-14 [19]	1081	1.79	1.5
D-15 [14]	1439	2.34	2

From TABLE 4 and 5 it is clearly shown that the area of proposed multiplier is improved by 30% while the power and delay are improved by 25%.

6 APPLICATION

The proposed designs are used for image multiplication. The resultant difference between the images multiplied by the exact multiplier to the approximate multiplier is given as mean square error (MSE). Based on the MSE, the best approximate multiplier quality is identified by the PSNR value of the multiplied images [36]. The calculation for MSE and PSNR is given in equation (5) and (6).

$$MSE = \frac{1}{mn} \sum_{i=1}^m \sum_{j=1}^n [X(i, j) - Y(i, j)]^2 \quad (5)$$

$$PSNR = 10 \log_{10} \left(\frac{MAX_i^2}{MSE} \right) \quad (6)$$

MAX_i gives the maximum possible values of each pixel, and image dimension is equal to m and n , where $X(i, j)$ and $Y(i, j)$ are the values of each pixel in i and j points of the accurate and non-accurate picture respectively. The PSNR of each noisy image is depicted in Fig. 10,11. The medical image are collected from 16-bit Digital Imaging and Communications in Medicine(DICOM) aycan OsiriXPRO (<http://www.aycan.de/lp/sample-dicom-images.html>). In

Fig. 10 MR image of LOC(right)

Fig.10 the 16-bit depth LOC(right) MR images are taken and the Contrast Ratio(CR) is varied through the proposed and existing multiplier to compare the various PSNR levels.

Fig. 11 CT image of Arterielle

In Fig.11 the CT image of Arterielle are multiplied with proposed and existing multiplier then the PSNR levels are compared.

7 MODEL PREDICTION

With the advancement in modern digital technology, machine learning and deep learning deals with big data to more precisely predict the weather forecast, Trade investment, hospital patient management ,etc. From (<https://www.aliza-dicom-viewer.com/download/datasets>) the big data-set is constructed with several subject. The model is developed with 62% of the initial data (training set) was used for training while 38% (testing set) was set aside randomly for post-training evaluation of our models. After multiplication with proposed and existing system the input features like nature of image, height, width, CR, PSNR are mapped on the excel-sheet to predict the best system as shown in Fig.12.

In the excel-sheet system 0-5 are mapped with design-10 to design-15, nature 0 stands for CT image of Arterielle. Like wise 15 subjects were taken. Each subjects consists of 20 images and all are multiplied with system 0-5, so the data-set is created with 1800 samples. The improved multi-class classification algorithm of LR and SVM in [34] are used to predict the best system. The training graph and confusion matrix shown in Fig.13. When the new medical image is taken for the contrast scaling depend on input parameter

	IMAGE NO	SIZE	NATURE	INPUT CR	PSNR	CR	system
0	1	0	0	1.000000	35.1	0.970940	0
1	1	0	0	0.772095	36.2	1.000000	1
2	1	0	0	0.783767	37.1	1.000000	2
3	1	0	0	0.766125	35.1	0.987057	3
4	1	0	0	0.763790	33.4	0.978253	4

Fig. 12 Input features created on Excel-sheet

Fig. 13 (a) Confusion matrix LR (b) Confusion matrix SVM (c) Training plot for LR (d) Training plot for SVM

the algorithm will identify the best approximate method with accuracy for SVM=96.52% and LR=95.65%.

8 CONCLUSION

The new architecture of scalable-split approximate compressor design has been done in this work. The multiplier constructed with new approximate compressors shows an average improvement of 25% of the area, 27% of power, and 20% time delay. The proposed multiplier produces the PSNR value of 5dB to 12dB higher than the previous design for the different application images. In 32x32, multiplier Design-3 produces attractive results, but in 16x16 multiplier, Design-9 is good in circuit aspects, and Design-7 is good in image quality. So depending on the target, the machine learning model identifies the

best multiplier. In future work the standard alone image processing processor are constructed with adaptive approximate computing to reduce the energy consumption.

References

1. S. M. Kuo, B. H. Lee, W. Tian, Real-time digital signal processing: fundamentals, implementations and applications, John Wiley & Sons, 2013.
2. A. Dandapat, S. Ghosal, P. Sarkar, D. Mukhopadhyay, A 1.2-ns16×16-bit binary multiplier using high speed compressors, *International Journal of Electrical and Electronics Engineering* 4 (3) (2010) 234–239.
3. K. K. Parhi, VLSI digital signal processing systems: design and implementation, John Wiley & Sons, 2007.
4. B. Parhami, Computer arithmetic, Vol. 20, Oxford university press, 2010.
5. S. Mittal, A survey of techniques for approximate computing, *ACM Computing Surveys (CSUR)* 48 (4) (2016) 1–33.
6. A. Gorantla, P. Deepa, Design of approximate subtractors and dividers for error tolerant image processing applications, *Journal of Electronic Testing* 35 (6) (2019) 901–907.
7. Y. Kim, Y. Zhang, P. Li, Energy efficient approximate arithmetic for error resilient neuromorphic computing, *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 23 (11) (2014) 2733–2737.
8. Y. Zhou, J. Lin, J. Wang, Z. Wang, Approximate comparator: Design and analysis, in: 2018 IEEE International Workshop on Signal Processing Systems (SiPS), IEEE, 2018, pp. 1–5.
9. M. Monajati, S. M. Fakhraie, E. Kabir, Approximate arithmetic for low-power image median filtering, *Circuits, Systems, and Signal Processing* 34 (10) (2015) 3191–3219.
10. A. Pishvaie, G. Jaberipur, A. Jahanian, Improved cmos (4; 2) compressor designs for parallel multipliers, *Computers & Electrical Engineering* 38 (6) (2012) 1703–1716.
11. C.-H. Chang, J. Gu, M. Zhang, Ultra low-voltage low-power cmos 4-2 and 5-2 compressors for fast arithmetic circuits, *IEEE Transactions on Circuits and Systems I: Regular Papers* 51 (10) (2004) 1985–1997.
12. A. Gorantla, Design of approximate compressors for multiplication, *ACM Journal on Emerging Technologies in Computing Systems (JETC)* 13 (3) (2017) 1–17.
13. M. H. Moaiyeri, F. Sabetzadeh, S. Angizi, An efficient majority-based compressor for approximate computing in the nano era, *Microsystem Technologies* 24 (3) (2018) 1589–1601.
14. M. Taheri, A. Arasteh, S. Mohammadyan, A. Panahi, K. Navi, A novel majority based imprecise 4: 2 compressor with respect to the current and future vlsi industry, *Microprocessors and Microsystems* 73 (2020) 102962.
15. S. Asif, Y. Kong, Low-area wallace multiplier, *Vlsi Design* 2014 (2014).
16. W. J. Townsend, E. E. Swartzlander Jr, J. A. Abraham, A comparison of dadda and wallace multiplier delays, in: *Advanced signal processing algorithms, architectures, and implementations XIII*, Vol. 5205, International Society for Optics and Photonics, 2003, pp. 552–560.
17. K. B. Jaiswal, N. Kumar, P. Seshadri, G. Lakshminarayanan, Low power wallace tree multiplier using modified full adder, in: 2015 3rd international conference on signal processing, communication and networking (ICSCN), IEEE, 2015, pp. 1–4.
18. T.-C. Lu, C.-Y. Tseng, C.-C. Huang, K.-M. Deng, 16-bit dicom medical images lossless hiding scheme based on edge sensing prediction mechanism, in: *Genetic and Evolutionary Computing*, Springer, 2015, pp. 189–196.
19. R. Marimuthu, Y. E. Rezinold, P. S. Mallick, Design and analysis of multiplier using approximate 15-4 compressor, *IEEE Access* 5 (2016) 1027–1036.
20. S. T. Bala, D. Shangavi, P. Sangeetha, Area and power efficient approximate wallace tree multiplier using 4: 2 compressors, in: 2018 International Conference on Intelligent Computing and Communication for Smart World (I2C2SW), IEEE, 2018, pp. 287–290.
21. S. Reda, M. Shafique, *Approximate Circuits*, Springer, 2019.

22. J. S. Walker, A primer on wavelets and their scientific applications, CRC press, 2008.
23. C. Sun, H. Talbot, S. Ourselin, T. Adriaansen, Digital Image Computing: Techniques and Applications: Proceedings of the VIIth Biennial Australian Pattern Recognition Society Conference, DICTA 2003, CSIRO PUBLISHING, 2003.
24. A. Abraham, X. H. Jiang, V. Snášel, J.-S. Pan, Intelligent Data Analysis and Applications: Proceedings of the Second Euro-China Conference on Intelligent Data Analysis and Applications, ECC 2015, Vol. 370, Springer, 2015.
25. P. Maragos, R. W. Schafer, M. A. Butt, Mathematical morphology and its applications to image and signal processing, Vol. 5, Springer Science & Business Media, 2012.
26. N. Cvejic, T. Seppänen, Digital audio watermarking techniques and technologies: applications and benchmarks, Information Science Reference Hershey, 2008.
27. Y. Guo, H. Sun, L. Guo, S. Kimura, Low-cost approximate multiplier design using probability-driven inexact compressors, in: 2018 IEEE Asia Pacific Conference on Circuits and Systems (APCCAS), IEEE, 2018, pp. 291–294.
28. C.-W. Tung, S.-H. Huang, Low-power high-accuracy approximate multiplier using approximate high-order compressors, in: 2019 2nd International Conference on Communication Engineering and Technology (ICCET), IEEE, 2019, pp. 163–167.
29. R. Marimuthu, D. Bansal, S. Balamurugan, P. Mallick, Design of 8-4 and 9-4 compressors for high speed multiplication, American Journal of Applied Sciences 10 (8) (2013) 893.
30. B. Silveira, G. Paim, B. Abreu, M. Grellert, C. M. Diniz, E. A. C. da Costa, S. Bampi, Power-efficient sum of absolute differences hardware architecture using adder compressors for integer motion estimation design, IEEE Transactions on Circuits and Systems I: Regular Papers 64 (12) (2017) 3126–3137.
31. M. Rouholamini, O. Kavehie, A.-P. Mirbaha, S. J. Jasbi, K. Navi, A new design for 7: 2 compressors, in: 2007 IEEE/ACS International Conference on Computer Systems and Applications, IEEE, 2007, pp. 474–478.
32. P. Radhakrishnan, G. Themozhi, Fpga implementation of xor-mux full adder based dwt for signal processing applications, Microprocessors and Microsystems 73 (2020) 102961.
33. P. Balasubramanian, N. E. Mastorakis, High speed gate level synchronous full adder designs, WSEAS Transactions on Circuits and Systems 8 (2) (2009) 290–300.
34. V. Arvind, J. S. Kim, E. K. Oermann, D. Kaji, S. K. Cho, Predicting surgical complications in adult patients undergoing anterior cervical discectomy and fusion using machine learning, Neurospine 15 (4) (2018) 329.
35. T. Schiavon, G. Paim, M. Fonseca, E. Costa, S. Almeida, Exploiting adder compressors for power-efficient 2-d approximate dct realization, in: 2016 IEEE 7th Latin American Symposium on Circuits & Systems (LASCAS), IEEE, 2016, pp. 383–386.
36. S. Venkatachalam, S.-B. Ko, Design of power and area efficient approximate multipliers, IEEE Transactions on Very Large Scale Integration (VLSI) Systems 25 (5) (2017) 1782–1786.

Funding: This research work received no external funding.

Acknowledgments: The authors acknowledge the VLSI design laboratory, Dept of Electrical and Computer Engineering, California State University, Fresno, USA. For providing the measurement facilities.

Availability of data and material: NIL

Code availability: NIL

Conflicts of Interest: The authors declare no conflict of interest.

Figures

Figure 1

(a) Full Adder (b) Xor mux

Figure 2

Salable Split compressor

Figure 3

Critical path in compressor (a) longest path (b) Splited path

Figure 4

Approximation finder

Figure 5

32x32 multiplier with exact scalable compressor

Figure 6

16X16 multiplier with split scalable compressor

Figure 7

16:2 split compressor

/e 12/cin6cout8	1572864	1572850	1572851	1572852	1572853
/e 12/cin7sum	1048576	1048569		1048570	
/e 12/cin7carry	1572864	1572850	1572851	1572852	1572853
/e 12/cin7cout0	1048576	1048568		1048569	
/e 12/cin7cout1	1048576	1048565	1048566		1048567
/e 12/cin7cout2	1048576	1048562	1048563	1048564	1048565
/e 12/cin7cout3	1048576	1048562	1048563	1048564	1048565
/e 12/cin7cout4	1048576	1048562	1048563	1048564	1048565
/e 12/cin7cout5	1048576	1048562	1048563	1048564	1048565
/e 12/cin7cout6	1048576	1048562	1048563	1048564	1048565
/e 12/cin7cout7	1048576	1048562	1048563	1048564	1048565
/e 12/cin7cout8	1048576	1048562	1048563	1048564	1048565
/e 12/cin8sum	1048576	1048569		1048570	
/e 12/cin8carrv	1572864	1572850	1572851	1572852	1572853
Now	2097153 ps	2097140 ps			
Cursor 3	2097152 ps				

Figure 8

Simulation Output for highly matched values

Figure 9

Histogram for NED of 16X16 multiplier

Figure 10

MR image of LOC(right)

CT image of Arterielle

	IMAGE NO	SIZE	NATURE	INPUT CR	PSNR	CR	system
0	1	0	0	1.000000	35.1	0.970940	0
1	1	0	0	0.772095	36.2	1.000000	1
2	1	0	0	0.783767	37.1	1.000000	2
3	1	0	0	0.766125	35.1	0.987057	3
4	1	0	0	0.763790	33.4	0.978253	4

Input features created on Excel-sheet

(a)

(c)

(b)

(d)

Figure 13

(a) Confusion matrix LR (b) Confusion matrix SVM (c) Training plot for LR (d) Training plot for SVM