IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

AN EFFICIENT VLSI DESIGN OF FIXED-WIDTH BOOTH MULTIPLIER FOR EDGE DETECTION SCHEME

M.Suganthi, Pg Scholar, Sir Issac Newton College Of Engineering And Technology

M.kavitha, Assistant Professor, Sir Issac Newton College Of Engineering And Technology

P.Arivazhagan , Assistant Professor, Sir Issac Newton College Of Engineering And Technology

***Abstract*—In this paper, The Sobel operator, sometimes called the Sobel–Feldman operator or Sobel filter, is used in image processing and computer vision, particularly within edge detection algorithms where it creates an image emphasizing edges. Edge detection is an essential process used to determine the object margins in most of the computer vision applications. Sobel edge detection algorithm, which is a simple method of edge detection, detects edges of various objects in an image. Real-time image applications need to be processed with large pixel data for a given time interval. I propose a data scaling technology for use in a low-error fixed-width Booth multiplier (FWBM) to reduce truncation errors and implement into Sobel Operator Application. The Proposed multipliers is done by Verilog HDL and Simulated by Modelsim 6.4 c and MATLAB and Synthesized by Xilinx tool and proposed system implemented in FPGA Spartan 3 XC3S 200 TQ-144. I introduce the Multiplier design into sobel operator edge detection process. So, Most of the VLSI architectures proposed for implementing sobel edge detection systems use FPGA, due to the parallel computing and reconfigurable feature.**

***Index Terms*—Fixed-width Booth multiplier (FWBM), Trun- cation error, Data scaling technology (DST), VLSI**

I. INTRODUCTION

Multipliers are widely used in digital signal processing (DSP) techniques, such as discrete cosine transform (DCT)[1] and fast Fourier transform [2], as well as in finite impulse response filters [3]. The need for a simple but accurate fixed- width multiplier for use in DSP systems has been a topic of discussion for many years.

Two of the most popular types of fixed-width multipliers are the Baugh–Wooley (BW) array multiplier [4]-[6] and the Booth multiplier [7]-[16]. The Booth encoder reduces the num- ber of truncated partial products, and therefore, the accuracy of Booth multipliers is higher than that of BW multipliers [12]. The compensation bias of truncated partial products is esti- mated through a simulation and the statistical probability of a low-error fixed-width Booth multiplier (FWBM). Simulation- based error compensation is an accurate method for use in FWBMs; however, its circuit design is time consuming. Jou *et al.* [7] extracted the statistical properties from a simulation and used linear regression to achieve constant compensation in FWBMs. Song *et al.* [8] used a curve-fitting technique to reduce the truncation error and achieve adaptive compensation. Similarly, an exhaustive simulation was employed in the circuit used in [9] to establish a comparison algorithm, and Wang *et al.* [10] used additional product information to improve the accuracy; however, establishing an exhaustive simulation is time consuming, particularly for long-width multiplication, al- though exhaustive simulation achieves accurate compensation. Probability methods for reducing the simulation time have been presented; these methods calculate the probability of an element being in the truncated partial products of mul- tipliers [11]-[16]. A circuit-generated constant compensation value called probabilistic estimation bias (PEB) was derived from theoretical computation in [11]. The generalized PEB (GPEB) method was used in [12] to obtain a more accurate estimation of the area penalty by using

information from more columns (w) in the truncation part; the GPEB circuit provides a suitable tradeoff between accuracy and area/power. An adaptive conditional-probability estimator (ACPE) was presented in [13] for use in compensated circuit design; the ACPE was shown to improve the accuracy of multipliers. The compensation method, called probability and computer simulation (PACS), in [14] combined simulation and statistical approaches to obtain high accuracy with a small area/delay penalty. However, much time is spent setting up some parts of the circuits in these hybrid methods. In [15], the multilevel conditional-probability (MLCP) method was adopted, in which conditional probability is more complex. Although a high accuracy can be achieved with the use of MLCP, the area cost increases. However, with this method, a signal-to-noise ratio (SNR) similar to the ideal SNR value of a post-truncated (P-T) FWBM can be obtained on the basis of the column information ($w = 3$).

In this paper, we propose a data scaling technology (DST)
that can be employed in all types of low-error FWBMs to im-prove their accuracy. The main idea is from the block floating point (BFP) technology [17]. In BFP, every block of incoming data has a joint scaling factor corresponding to the data sample with the highest magnitude in the block, and the scaling idea is adopted in this work. The proposed DST uses the redundant bits of the multiplicand to more efficiently obtain bits for low-error FWBMs. More specifically, this method reduces the truncated partial products, which are used to estimate the compensation bias; however, the calculations in the FWBM are increased. Nevertheless, the accuracy of the FWBM can be improved by adding the proposed DST circuit to it. The results indicate that the use of the proposed DST increases the accuracy of low-error FWBMs with only a small area overhead. The proposed DST-FWBM was also implemented in a VLSI chip, with 0.18-μm CMOS technology used to demonstrate its performance. The proposed DST method was shown to considerably improve the accuracy of all types of FWBMs and is thus suitable for use in applications that require high precision such as image processing.

This rest of the paper is organized as follows. Section II introduces the mathematical derivation of the low-error FWBM. Section III presents the proposed DST, the results of the accuracy analysis, and the result demonstrating the improvement in low-error FWBMs. Section IV presents the sobel edge detection. The architecture of the DST-FWBM is presented in Section V. Comparisons and discussions are presented in Section VI, and the conclusions are provided in Section VII.

II. LOW-ERROR FIXED-WIDTH BOOTH MULTIPLIER

In general, the 2L-bit product P can be expressed using a two's complement representation.

$$X = -x_{L-1}2^{L-1} + \sum_{i=0}^{L-2} x_i \cdot 2^i$$
$$Y = -y_{L-1}2^{L-1} + \sum_{i=0}^{L-2} y_i \cdot 2^i$$
$$P = X \times Y$$

where X and Y indicate the multiplier and multiplicand, respectively. In Booth encoding, to reduce the number of partial products [18], three concatenated inputs y2i+1, y2i , and y2i−1 are mapped to y ′ i [13]. The nonzero code nzi is a 1-bit digit, the value of which depends on whether y ′ i is equal to zero. Thus, the rows in the partial product array can be reduced to Q = L/2, where L is even. Based on the encoding of y ′ i , the partial products of an L × L FWBM can be illustrated as shown in Fig. 1. The partial product array can be divided into two parts: the main part (MP), which includes the most significant L columns, and the truncation part (TP), which includes the least significant L columns; the product P can be expressed as follows:

$$P = MP + TP. \quad (2)$$

For the fixed-width multiplication, the product P is truncated to P_q.

$$P \approx P_q = MP + TP = MP + \sigma \cdot 2^L \quad (3)$$

where σ represents the compensation bias of the low-error FWBM, which can be decomposed further into TP_{ma} and TP_{mi} by using the following equation:

$$\sigma = TP_{ma} + TP_{mi} \quad (4)$$

where indicates the floor operation. TP_{ma} and TP_{mi} are the major and minor compensation parts in the TP, respec- tively. TP_{ma} provides true information, and TP_{mi} provides compensation bias to the MP based on all types of compensation algorithms. Therefore, the compensation bias σ can be calculated by obtaining TP_{ma} and TP_{mi}. In many studies, TP_{mi} is estimated via probability [12], conditional probability [13], PACS [14], and MLCP [15] to keep the truncation error as low as that of a P-T FWBM. In general, a long-width FWBM has been mostly used in recent applications. Owing to the time-consuming simulation for a long-width FWBM, $L = 16$ is often used for discussion. The column information w has been discussed in previous studies, mainly with $w = 1$, $w = 2$, and $w = 3$

because the accuracy with w = 1 3 varies significantly

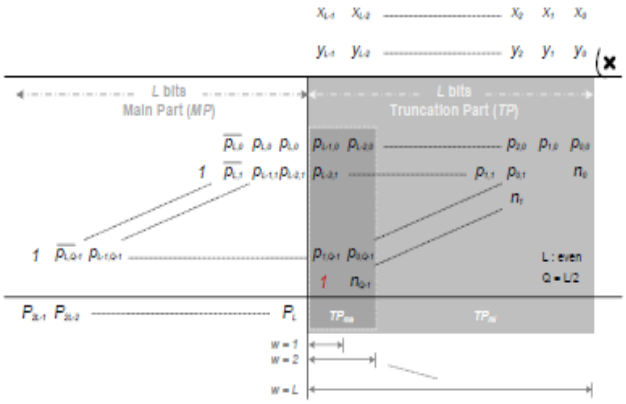

Fig 1: structure of FWBM

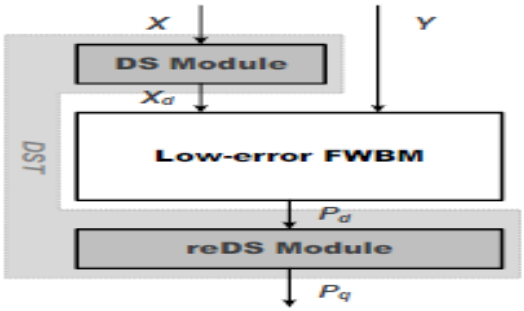

Fig 2: Low error FWBM

III. DATA SCALING TECHNOLOGY

The proposed DST method is incorporated as an additional circuit in low-error FWBMs. Thus, the circuit performance of the FWBMs is retained, with the DST circuit improving the accuracy of the low-error FWBM. In DST, the redundant bits of the multiplicand are employed to obtain more information from the truncated TP_{mi}. Fig. 2 presents the proposed DST to be used in low-error FWBMs. There are two DST Modules (**DS Module** and **reDS Module**). One of them is inserted above the low-error FWBM, and the other is inserted below the FWBM. The **DS Module** shifts the *Scal* bit according to the redundant bit of the multiplicand X, and the **reDS Module** reconstructs the results according to the corrected weight. The following equations express the operations of the **DS** and **reDS**

Modules:

$$X_d = X << Scal \quad (5)$$

$$P_q = P_d >> Scal \quad (6)$$

remove the redundant multiplicand X. For example, if L = 8, X = 8′b0000 1010 and *Scal* = 3, then X_d = 8′b0101$_0$000. where *Scal* indicates the number of bits to be shifted to Fig. 2 shows more DST operation examples. The *Scal* values are calculated using the number of redundant bits in the MSBs of the multiplicand X. In general, the range of *Scal* is [0, L-1]. However, a high value of *Scal* requires numerous bits to be shifted, which implies that more calculations and an increased circuit area would be required. Thus, a limited shift bits (DSb) factor is introduced to limit the value of *Scal*;

TABLE I

SNR VALUES VERSUS DS_B FOR DT, GPEB, ACPE, PACS, AND MLCP FWBMS WITH L = 16

Methods	D-T	GPEB			ACPE			PACS			MLCP		
		w = 1	w = 2	w = 3	w = 1	w = 2	w = 3	w = 1	w = 2	w = 3	w = 1	w = 2	w = 3
DSb 0	64.8	79.3	83.7	85.0	81.8	84.2	85.1	81.8	84.2	85.1	81.3	84.2	85.1
1	67.1	80.8	84.3	85.2	82.9	84.7	85.3	82.9	84.7	85.3	82.4	84.7	85.3
2	67.3	81.0	84.4	85.2	83.0	84.7	85.3	83.0	84.7	85.3	82.6	84.7	85.3
3	67.4	81.0	84.4	85.2	83.0	84.7	85.3	83.0	84.7	85.3	82.6	84.7	85.3
4	67.4	81.0	84.4	85.2	83.0	84.7	85.3	83.0	84.7	85.3	82.6	84.7	85.3
5	67.4	81.0	84.4	85.2	83.0	84.7	85.3	83.0	84.7	85.3	82.6	84.7	85.3
6	67.4	81.0	84.4	85.2	83.0	84.7	85.3	83.0	84.7	85.3	82.6	84.7	85.3
7	67.4	81.0	84.4	85.2	83.0	84.7	85.3	83.0	84.7	85.3	82.6	84.7	85.3
8	67.4	81.0	84.4	85.2	83.0	84.7	85.3	83.0	84.7	85.3	82.6	84.7	85.3
9	67.4	81.0	84.4	85.2	83.0	84.7	85.3	83.0	84.7	85.3	82.6	84.7	85.3
10	67.4	81.0	84.4	85.2	83.0	84.7	85.3	83.0	84.7	85.3	82.6	84.7	85.3
11	67.4	81.0	84.4	85.2	83.0	84.7	85.3	83.0	84.7	85.3	82.6	84.7	85.3
12	67.4	81.0	84.4	85.2	83.0	84.7	85.3	83.0	84.7	85.3	82.6	84.7	85.3
13	67.4	81.0	84.4	85.2	83.0	84.7	85.3	83.0	84.7	85.3	82.6	84.7	85.3
14	67.4	81.0	84.4		83.0	84.7		83.0	84.7		82.6	84.7	
15	67.4	81.0			83.0			83.0			82.6		
16	67.4												

however, this would also limit the extent to which the accuracy of the FWBMs can be improved. Thus, there is a tradeoff between accuracy and circuit area. The SNR is used to analyze the effect of DSb on accuracy. The SNR is defined as follows:

$$\text{SNR (dB)} = 10 \log \{E[P^2] / E[(P-Pq)^2]\} \quad (7)$$

where E [ ] is the expectation operation. Table I lists the SNR FWBMs (L = 16). When DSb = 0, DST is not applied to these FWBMs. Thus, the SNR values for DSb = 0 are the values obtained for various DSb values for existing low-error same as those in the original FWBMs. As the DSb increases, the SNR value increases, indicating that using a DSb increases the SNR. However, for higher DSb values, the increases in the corresponding SNR values are negligible. As an example, we have listed the increase in SNR values for 0 Dsb<<15 of the GPEB FWBM (w = 1) in Table II. The results show that the increase in SNR values becomes negligible as the DSb increases, which implies that if the DSb is extremely high, there is no increase in the SNR. According to the binary(#RBs) can be expressed as #RB = $2^L/2^{DSb} = 2^{L-DSb}$. The numbers shown in Table II, the number of redundant bits additional contribution of increasing DSb to the compensation bias becomes increasingly small, as expressed in (6)

IV. PROPOSED DST-FWBM CIRCUIT

For the VLSI design, the complexity and circuit area depend on the DSb. Thus, in this study, the tradeoff between accuracy and area cost was considered. As shown in Fig. 2, the proposed **DS Module** can be implemented using L D-to-1 multiplexers (MUXs) in which D equals (DSb +1), and the **reDS Module** is designed using (L-1) D-to-1 MUXs. Thus, the proposed DST circuit uses (2L- 1) D-to-1 MUXs, and the area of the low-error FWBM and the (2L-1) D-to-1 MUXs. To the entire DST-FWBM circuit is the sum of the area of evaluate the area overhead and accuracy, the area of a 3-to-1 MUX is assumed to be twice that of a 2-to-1 MUX, a 2-to-1 MUX, and so forth. Thus, a (D-1)-fold increase the area of a 4-to-1 MUX is assumed to be thrice that of in area is required for a D-to-1 MUX over a 2-to-1 MUX.

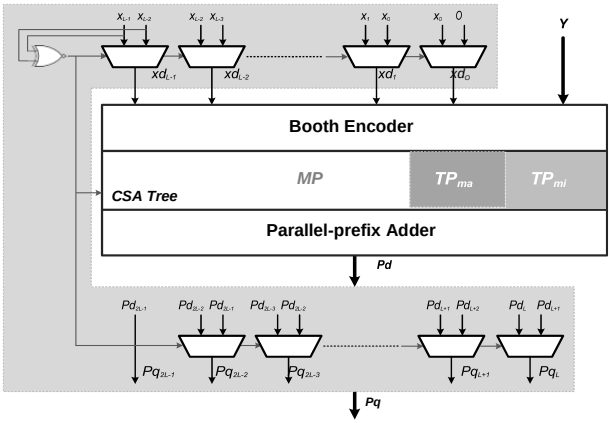

Fig. 3. Architecture of proposed DST-FWBM with DSb = 1.

TABLE II
INCREASE IN SNR VERSUS INCREASE IN DSB FOR A GPEB FWBM
(L = 16, w = 1)

DST-bit (DSb)	SNR Increased (dB)	#RB	
0	0	0	
1	1.427808548635	32768	$2^L/2^1$
2	0.232373350716	16384	$2^L/2^2$
3	0.030027676364	8192	$2^L/2^3$
4	0.003872098504	4096	$2^L/2^4$
5	0.000484576000	2048	$2^L/2^5$
6	0.000062295906	1024	$2^L/2^6$
7	0.000007805088	512	$2^L/2^7$
8	0.000001039949	256	$2^L/2^8$
9	0.000000134896	128	$2^L/2^9$
10	0.000000021163	64	$2^L/2^{10}$
11	0.000000003100	32	$2^L/2^{11}$
12	0.000000000597	16	$2^L/2^{12}$
13	0.000000000078	8	$2^L/2^{13}$
14	0.000000000011	4	$2^L/2^{14}$
15	0.000000000001	2	$2^L/2^{15}$

The proposed DST area is multiplied by the DSb value. However, according to the analysis of the increase in the SNR, as presented in Table II, the SNR does not continue to improve significantly when multiplied by the DSb value, and an improvement gap is evident between DSb= 0 and DSb= 1. Therefore, in this study, we chose DSb= 1 because this value provides maximum cost efficiency for the proposed DST. The architecture of the proposed DST-FWBM circuit is shown in Fig. 3. However, the DSb increases when the proposed DST is used, and a high SNR value is achieved. As indicated in Fig. 3, the proposed DST-FWBM consists of a DST circuit and a low-error FWBM, which could be a GPEB FWBM, ACPE FWBM, PACS FWBM, or MLCP FWBM. The DST circuit is implemented using (2L 1) 2-to- 1 MUXs with texttDSb = 1. The FWBM is surrounded by the DST circuit and consists of a Booth encoder, a carry-save adder (CSA) tree, and a parallel prefix adder. The CSA, which consists of either full adders or half adders, adds the partial products from the MP, TP_{ma}, and the estimated TP_{mi} in all the low-error FWBMs tested. Finally, the high-speed parallel prefix adder calculates the products P_d, and the final results P_q are obtained by using the DST circuit.

TABLE III
COMPARISON OF ACCURACY AND CIRCUIT PERFORMANCE OF DST-FWBM AND ORIGINAL FWBM
(L = 16)

Methods			$\bar{ç}$	$\overline{\lvert ç \rvert}$	Çmax	*SNR*	*MSE*	Area (μm²)	Delay (*ns*)	Power (*m*W)
P-T			0.25	0.00	0.50	85.5	0.084	1406	2.20	2.19
D-T		Orig.	3.00	3.00	8.00	64.8	9.876	630	1.59	0.97
		wl DST	2.13	2.13	8.00	67.1	5.830	729	1.83	1.27
w = 1	GPEB [12]		-0.25	0.48	2.50	79.3	0.305	691	1.69	1.07
		wl DST	-0.19	0.40	2.50	80.8	0.248	786	1.83	1.38
	ACPE [13]		-0.12	0.36	2.17	81.8	0.197	745	1.63	1.15
		wl DST	-0.09	0.32	2.17	82.9	0.153	840	1.83	1.45
	PACS [14]		-0.06	0.36	2.17	81.8	0.197	738	1.63	1.14
		wl DST	-0.06	0.32	2.17	82.9	0.153	833	1.83	1.44
	MLCP [15]		0.13	0.38	2.50	81.3	0.221	737	1.65	1.13
		wl DST	0.09	0.33	2.50	82.4	0.172	832	1.79	1.44
	Song [8]		0.24	0.40	1.96	80.9	0.241	808	1.85	1.27
	Wang [10]		0.00	0.36	2.17	81.8	0.196	745	1.65	1.13
w = 2	GPEB [12]		0.03	0.29	1.50	83.7	0.127	805	1.73	1.28
		wl DST	0.02	0.28	1.50	84.3	0.111	901	1.94	1.61
	ACPE [13]		-0.06	0.28	1.25	84.2	0.113	807	1.69	1.26
		wl DST	-0.05	0.27	1.25	84.7	0.101	902	1.88	1.58
	PACS [14]		0.03	0.28	1.25	84.2	0.113	802	1.77	1.24
		wl DST	0.02	0.27	1.25	84.7	0.101	879	1.90	1.57
	MLCP [15]		0.06	0.28	1.25	84.2	0.113	807	1.69	1.25
		wl DST	0.05	0.27	1.25	84.7	0.101	903	1.88	1.57
	Song [8]		0.12	0.28	1.14	84.0	0.118	862	1.85	1.37
w = 3	GPEB [12]		0.02	0.26	1.00	85.0	0.094	865	1.80	1.38
		wl DST	0.01	0.26	1.00	85.2	0.090	960	2.00	1.72
	ACPE [13]		-0.03	0.26	0.88	85.1	0.092	907	1.77	1.44
		wl DST	-0.02	0.26	0.88	85.3	0.088	1002	1.96	1.78
	PACS [14]		0.02	0.26	0.88	85.1	0.092	900	1.77	1.43
		wl DST	0.01	0.26	0.88	85.3	0.088	995	1.96	1.77
	MLCP [15]		0.03	0.26	0.88	85.1	0.088	911	1.93	1.44
		wl DST	0.02	0.26	0.88	85.3	0.088	1006	2.06	1.78
	Song [8]		0.01	0.26	0.82	85.2	0.090	924	1.88	1.48

V. SOBEL EDGE DETECTION

Edge detection algorithms are widely used in various research fields like Image Processing, Video Processing and Artificial Intelligence etc. Edges are most important attribute of image information and a lot of edge detection algorithms are defined in literature. Sobel edge detection algorithm is chosen among of them due to its property of less deterioration in high level of noise. FPGA is becoming the most dominant form of programmable logic over past few years and it has advantages of low investment cost and desktop testing with moderate processing speed and thereby offering itself as suitable one for real time application. This paper describes an efficient architecture for Sobel edge detector which is faster and takes less space than the Existing architecture.

1- First order derivative operator's (gradient method) is contain Robert Detector, Prewitt Detector and Sobel Detector where: Robert Detector : It is gradient based operator. It firstly computes the sum of the squares of the difference between diagonally adjacent pixels through discrete differentiation and then calculate approximate gradient of the image. The input image is convolved with the default kernels of operator and gradient magnitude and directions are computed. It uses following 2x2 two kernels as shown in Gx,Gy

Gx=

-1	0
0	1

Gy=

0	-1
1	0

Prewitt Detector: The function of Prewitt edge detector is almost same as of Sobel detector but have different kernels as shown

Gx=

-1	0	1
-1	0	1
-1	0	1

Gy=

1	1	1
0	0	0
-1	-1	-1

Sobel Detector is one of the most frequently used in edge detection . Sobel edge detection can be implemented by filtering an image with left mask or kernel. Filter the image again with the other mask. After this square of the pixels values of each filtered image. Now add the two results and compute their root. The 3 × 3 convolution masks for the Sobel based operator as shown

Gx=

-1	-2	-1
0	0	0
1	2	1

Gy=

-1	0	1
-2	0	2
-1	0	1

Sobel has two main advantages: it has some smoothing effect to the random noise of the image:

1) Since the introduction of the average factor, it has some smoothing effect to the random noise of the image.

2) Because it is the differential of two rows or two columns, so the element of the edge on both sides ha been enhanced, so that the edge seems thick and bright. The Sobel operator is used mostly although it is slower than the Roberts cross operator, because its horizontal and vertical kernels smooth the input image and makes operator less sensitive to noise. The reason for using Sobel operator is that it has relatively small masks compare to other operators.

2- Second order derivative operator's is contain Laplacian of Gaussian:

The Laplacian is a 2-D isotropic measure of the 2nd spatial 4derivative of an image. The Laplacian of an image highlights regions of rapid intensity change and is therefore often used for edge detection. The operator normally takes a single gray level image as input and produces another gray level image as output. The Laplacian L(x,y) of an image with pixel intensity values I(x,y) is given below:

$$L(x,y) = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2}$$

Since the input image is represented as a set of discrete pixels, we have to find a discrete convolution kernel that can approximate the second derivatives in the definition of the Laplacian. Three commonly used small kernels are shown in below

0	1	0
1	-4	1
0	1	0

1	1	1
1	-8	1
1	1	1

-1	2	-1
2	-4	2
-1	2	-1

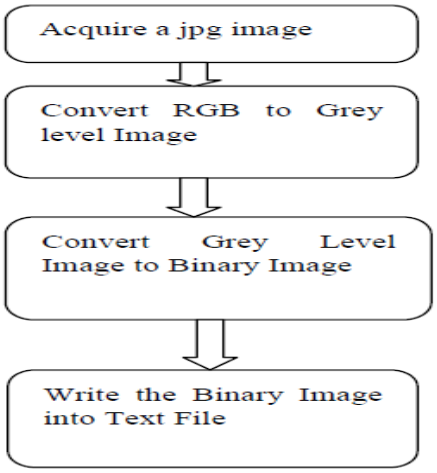

Fig 4: Edge detection design flow

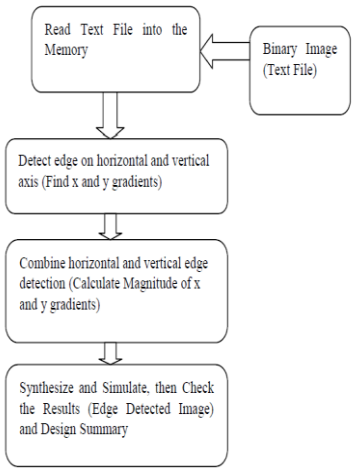

Fig 5: Flow chart

VI. Implementation Results and Discussion

A. Comparison with Original FWBM

To evaluate the accuracy problem, the average error ε, absolute average error ε , maximum absolute error ε_{max}, mean squared error (MSE), and SNR of the proposed DST-FWBM multiplier with D-T, P-T, GPEB, ACPE, PACS, and MLCP for L = 16 are compared; the results of these comparisons are listed in Table III. The terms ε, ε , ε_{max}, and MSE are defined as follows:

$$\varepsilon = E[P-P_q]/2^L \quad (8)$$

$$|\varepsilon| = E[|P-P_q|]/2^L \quad (9)$$

$$\varepsilon_{max} = \max\{|P-P_q|\}/2^L \quad (10)$$

$$MSE = E[(P-P_q)^2]/2^2 \quad (11)$$

Where and max indicate the absolute value and maximum operations, respectively. The SNR is expressed in (7). The data in Table III indicate that a clear improvement in accuracy was achieved when the proposed DST method was used with the low-error FWBMs. The SNR value is a critical factor in system application; the proposed DST circuit improved the SNR of the DT FWBM, GPEB FWBM [12], ACPE FWBM [13], PACS FWBM [14], and MLCP FWBM [15] by 3.51%, 1.8%, 1.28%, and 1.38%, respectively. Furthermore, compared with the FWBMs proposed in [8] and [10], the proposed DST-FWBM achieved higher SNR and method [10] cannot apply to $w > 1$ with a closed-form, thus, there are no results for [10] in the case of $w = 2$ and $w = 3$. lower MSE values. Note that, using an exhaustive simulation The SNR value of the MLCP with DST (w = 3, DSb = 1) was 85.28 dB, which is similar to the SNR value of the golden P-T FWBM; the difference between their SNR values is only 0.25 dB. In this case, the SNR value is the largest in Table III. However, the εmax values were the same as those of the original low-error FWBMs. The proposed DST method did not reduce εmax, but existing methods that can handle large values of w can reduce εmax. Furthermore, the DST circuit increased the area and delay at higher w values, as shown in Table III. The proposed method, however, does not seem to yield results superior to those of existing methods by using information from more columns, but it can improve the accuracy of the original FWBMs; moreover, efficiency can be improved by using more column information. If the original FWBM circuit cannot be changed, then column information is not added, which means that the accuracy cannot be improved. In such a case, it is appropriate to apply the DST.

B. Evaluation of the DST Circuit in Actual Applications

To verify the suitability of the use of the proposed DST circuit in actual applications, the DST-FWBM was applied to two-dimensional discrete cosine transform (2-D DCT) [19]. accuracy; all images had a resolution of 512 512 pixels, Ten popular test images were used to evaluate the system with each pixel represented by 8-bit 256 gray-level data. The system accuracy is defined by the peak SNR (PSNR), and a comparison of the 2-D DCT core with the applied conventional low-error FWBMs is presented in Table IV, which also presents data on circuit performance. The Synopsys Design Compiler was used to synthesize the RTL design of the DCT cores. TSMC 40-nm standard cells were used to synthesize the design, and the proposed 2-D DCT core was operated at 100 MHz. The area of the core for all FWBMs is presented in Table IV. The results show the FWBM with proposed DST can improve the PSNR value compared to that FWBM with DST (w = 3) achieved high PSNR values that were only 0.1% less than those of the P-T FWBM, but the without the DST circuit significantly. The

proposed MLCP area of the MLCP FWBM with DST was 9.1% smaller than that of the P-T FWBM.

C. Result

Finally, the edges can be detected by applying the threshold by using equation (5) to the total gradient (Gr). If (Gr) is greater than the threshold, then pixel should be identified an edge as shown. Else it's not identified as an edge. This Edge Detection logic is made by Schostic Logic Circuit.

Fig 6: Edge detection

VII. CONCLUSION

In this study, we presented the application of a DST circuit to low-error FWBMs; the proposed DST circuit considerably improved the accuracy of the FWBMs. The accuracy of the proposed DST-FWBM was close to the ideal accuracy value of P-T multipliers and exhibited a justifiably small area cost. Upon evaluation of its system application, the proposed DST- FWBM achieved high accuracy. The DST circuit also helped improve the accuracy of long-width FWBMs. In summary, DST can be used in Edge detection scheme, particularly those that require high accuracy.

REFERENCES

[1] Y. H. Chen, T. Y. Chang, and C. Y. Li, "High Throughput DA-based DCT with High Accuracy Error-compensated Adder Tree," *IEEE Trans. VLSI Syst.*, vol. 19, no. 4, pp. 709–714, Apr. 2011.

[2] S. N. Tang, J. W. Tsai, and T. Y. Chang, "A 2.4-GS/s FFT Processor for OFDM-Based WPAN Applications," *IEEE Trans. Circuits Syst. II*, vol. 57, no. 6, pp. 451–455, Jun. 2010.

[3] W. Liu, T. Cao, P. Yin, Y. Zhu, C. Wang, E. E. Swartzlander, and F. Lombardi, "Design and analysis of approximate redundant binary multipliers," *IEEE Trans. Comput.*, vol. 68, no. 6, pp. 804–819, Jun. 2019.

[4] L. D. Van, S. S. Wang, and W. S. Feng, "Design of the lower error fixed-width multiplier and its application," *IEEE Trans. Circuits Syst. II*, vol. 47, no. 10, pp. 1112–1118, Oct. 2000.

[5] I. C. Wey and C. C. Wang, "Low-error and hardware-efficient fixed- width multiplier by using the dual-group minor input correction vector to lower input correction vector compensation error," *IEEE Trans. VLSI Syst.*, vol. 20, no. 10, pp. 1923–1928, Oct. 2012.

[6] D. Esposito, A. G. M. Strollo, E. Napoli, D. De Caro, and N. Petra, "Approximate multipliers based on new approximate compressors," *IEEE Trans. Circuits Syst. I*, vol. 65, no. 12, pp. 4169–4182, Dec. 2018.

[7] S. J. Jou, M. H. Tsai, and Y. L. Tsao, "Low-error reduced-width Booth multipliers for DSP applications," *IEEE Trans. Circuits Syst. I*, vol. 50, no. 11, pp. 1470–1474, Nov. 2003.

[8] M. A. Song, L. D. Van, and S. Y. Kuo, "Adaptive low-error fixed- width Booth multipliers," *IEICE Trans. Fundamentals*, vol. E90-A, no. 6, pp. 1180–1187, Jun. 2007.

[9] Y. H. Chen, T. Y. Chang, and R. Y. Jou, "A statistical error-compensated Booth multiplier and its DCT applications," in *Proc. IEEE Region 10 Conf.*, 2010, pp. 1146–1149.

[10] J. P. Wang, S. R. Kuang, and S. C. Liang, "High-Accuracy Fixed-Width Modified Booth Multipliers for Lossy Applications," *IEEE Trans. VLSI Syst.*, vol. 19, no. 1, pp. 52–60, Jan. 2011.

[11] C. Y. Li, Y. H. Chen, T. Y. Chang, and J. N. Chen, "A probabilistic estimation bias circuit for fixed-width Booth multiplier and its DCT applications," *IEEE Trans. Circuits Syst. II*, vol. 58, no. 4, pp. 215–219, Apr. 2011.

[12] Y. H. Chen, C. Y. Li, and T. Y. Chang, "Area-Effective and Power- Efficient Fixed-Width Booth Multipliers Using Generalized Probabilistic Estimation Bias," *IEEE J. Emerging Sel. Topics Circuits Syst.*, vol. 1, no. 3, pp. 277–288, Sep. 2011.

[13] Y. H. Chen and T. Y. Chang, "A High-Accuracy Adaptive Conditional- Probability Estimator for Fixed-width Booth Multipliers," *IEEE Trans. Circuits Syst. I*, vol. 59, no. 3, pp. 594–603, Mar. 2012.

[14] W. Q. He, Y. H. Chen, and S. J. Jou, "High-Accuracy Fixed-Width Booth Multipliers Based on Probability and Simulation," *IEEE Trans. Circuits Syst. I*, vol. 62, no. 8, pp. 2052–2061, Aug. 2015.

[15] Y. H. Chen, "An Accuracy-Adjustment Fixed-Width Booth Multiplier Based on Multilevel Conditional Probability," *IEEE Trans. VLSI Syst.*, vol. 23, no. 1, pp. 203–207, Jan. 2015.

[16] Yuan-Ho Chen,"Improvement of Accuracy of Fixed-Width Booth Multipliers Using Data Scaling Technology", IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS-II, Jan 2020.

[17] Z. Zhang and Y. He, "A Low-Error Energy-Efficient Fixed-Width Booth Multiplier With Sign-Digit-Based Conditional Probability Estimation," *IEEE Trans. Circuits Syst. II*, vol. 65, no. 2, pp. 236–240, Feb. 2018.

[18] M. Chakraborty, R. Shaik, and M. H. Lee, "A Block-Floating-Point- Based Realization of the Block LMS Algorithm," *IEEE Trans. Circuits Syst. II*, vol. 53, no. 9, pp. 812–816, 2006.

[19] B. Parhami, *Computer arithmetic: algorithms and hardware designs*. Oxford, UK: Oxford University Press, 2000.

[20] S. C. Hsia and S. H. Wang, "Shift-register-based data transposition for cost-effective discrete cosine transform," *IEEE Trans. VLSI Syst.*, vol. 15, no. 6, pp. 725–728, Jun. 2007.