Hardware Implementation of Robust Dwt-Dht Watermark on Fpga

Pankaj U. Lande*, Sanjay N. Talbar**, G.N. Shinde*
* Dept. Electronics, Rajaram College, Kolhapur
** S.G.G.S. Institute of Engineering and Technology
*Pro-vice Chancellor S.R.T.M. University, Nanded

Article Info

Keyword:

Watermarking
Field Programmable Gate Array
Cohen-Daubechies-Feauveau
5/3 Wavelet
JPEG 2000
Discrete Hadamard Transform

ABSTRACT

This paper presents a novel hardware architecture for watermarking implemented using Discrete Wavelet Transform (DWT) and Discrete Hadamard Transform (DHT). The algorithm utilises the space frequency property of wavelet transform and frequency separation property of DHT. We have used the Cohen-Daubechies-Feauveau (CDF5/3) wavelet which is the basis of loss less JPEG2000 compression standard. The objective was to combine two different transforms to compensate their disadvantages and make the watermarking technique more robust. We have used serial-parallel combination approach which gives the best balance between speed and hardware utilisation. This real-time low cost and robust watermarking hardware solution can be incorporated into existing digital imaging systems such as digital camera, scanners, camcorders etc. To the best of our knowledge, this is the first attempt of implementing the dual transform for watermarking on hardware platform as well as for DWT-DHT combined approach.

Corresponding Author:

Dr. Pankaj U. Lande,
Department of Electronics, Rajaram College,
Kolhapur, Maharashatra, India -411007
Email: landepankaj@gmail.com

1. INTRODUCTION

In the recent advances in technology and success of the internet brought a new problem like *digital piracy*. The user can now download free software to modify the digital content. This software is used as a tool to modify the content and share it over the internet with the lighting speed. Distribution of digital content such as movies, music, and images is now faster and easier via computer technology, especially on the Internet[1]. These digital contents are shared on the internet without the owner's concern[2].

Digital Right Management (DRM) is a collection of techniques that enables the licensing of digital content such as image, video and music, digital documents, eBooks etc. DRM consist of two prominent technologies. They are encryption and watermarking. Watermarking is a technique used for paper manufacturing is now adopted by the digital world coined as *Digital Water*marking. In watermarking processes, an information signal called as the watermark is embedded into multimedia objects such as an image to protect the ownership rights [3][4]. Today, the watermarking technology not only has copyright application but it is used in various applications. Few of them are listed below: -

- ✓ **Medical Applications:** Names of the patients can be incorporated in the X-ray reports and MRI scans using a visible watermarking technique[5].
- ✓ **Broadcast Monitoring and Tracking:** Digital watermarks are used to track a broadcast of media over a channel for various purposes. Such as advertisers can actually pay for only the number of times the advertisement was actually relayed on broadcast [6].
- ✓ **Fingerprinting: -** The copies can be made available with the unique sequence of numbers as a watermark. If a copy is made illegally the source can be easily tracked with its unique sequence number[7].

Watermarking techniques can be divided into various categories in a number of ways. According to watermark extraction or detection, the watermarking is categorised into Non-blind, Semi-Blind and Blind schemes [8].Watermarking can be done in spatial or transformed domain using transformed like Discrete Cosine Transform (DCT)[9], Discrete Wavelet Transform [10], Discrete Hartely Transform (DHT)[11]. Spatial domain method provides an algorithm that directly operates on the pixel values of the host image [12].

1.1 NECESSITY FOR HARDWARE IMPLEMENTATION

The watermarking algorithm can be implemented using the software or the hardware[13][14]. Designing a hardware for the watermarking have many advantages such as the hardware can be used in imaging systems such as digital camera, video camcorders and scanners. This will fill the gap between the capturing an image and then putting the watermark insides using the software. The recent advances in FPGA technology are fast and it consumes less power. Using present technology, the watermarking algorithm can be implemented for real time and battery operated applications. This hardware approach can be tested and verified to implement on System on Chip (SoC) for large scale of production. The bulk production of the integrated circuits will make it at a lower cost than the software. The following section discusses the hardware implementation of image watermarking algorithms presented in various literature.

2. BACK GROUND

There are many ways to embed a watermark. Most of them are based on single transform. There are some literature publishing the dual transforms based watermarking technique. Among them, DWT-DCT is the most common approach. The dual transforms are used in different ways such as apply of the first transform and apply the second transform on the partial part of the first one. Another way is to transform the image using one transform and for the watermark, another transform is used. The basic purpose of the using dual transform is to make watermarking technology more robust.

2.1 Related Work

In software, there are many kinds of literature presented for watermarking using dual transform. Most of them are using DWT -DCT combination. In[15], Afroja Akter et al. presented a dual watermarking technique based on DWT and DCT. The original image is transformed into wavelet domain up to four levels. The vertical and horizontal components are further divided into blocks. These blocks are then DCT transformed to embed a watermark. The process is repeated until the second level. The watermarking algorithm uses the average value to calculate the embedding strength of the watermark. The algorithm performs well against the additive white Gaussian noise attack.

Hwai-Tsu Hu and et al. presented an adaptive watermarking algorithm based on the quantization index modulation is described in [16]. The cover image is transformed into wavelet domain. The low-frequency LL was selected for the watermarking purpose. This frequency band was further divided into blocks and DCT was computed to find out the higher frequency components. These higher frequencies are used to embed a binary watermark. The watermark is scrambled by using Arnold transform to enhance security. The watermarking algorithm gives a good performance against JPEG attack and the other geometric attacks. A similar approach described in [17]. The host image is transformed into wavelet domain. DWT further applied to the HL or HH band to obtain further detail components. These higher frequency components are further divided into equal size blocks to compute DCT. A binary watermark is embedded by modifying the middle frequency of the DCT transformed block. A pair of PN sequence is used with fixed gain to embed the binary pattern. the watermarking schemes perform well against the geometric as well as the compression attacks.

In [18], a combined approach of DWT-DCT and SVD is used. The host image is transformed into wavelet domain. In transformed image LL subband and watermarked image is then DCT transformed. Further, the SVD is calculated and the S components are modified to embed the watermark. This scheme shows good results against the non-geometric attacks. Another very similar approach has been proposed in[19]. In the proposed algorithm, DWT – SVD and then DCT is used. The algorithm shows good performance against the compression attacks as well as geometric attacks.

Hardware implementations are presented in the following literature. In[20], spatial domain watermarking method is described. The method is a combination of the stereoscope image compression and watermarking algorithm. A parallel architecture was designed to implement a watermark. Least significant bit of disparity value is replaced by watermark bit. The algorithm was implemented on Xilinx Virtex IV technology.

A hardware implementation of wavelet domain watermarking algorithm is illustrated in [21].The cover image is transformed into wavelet domain using 5/3 LeGal filters up to three levels. The LL3 band is selected for watermarking. These pixels are further divided into blocks of eight pixels. The difference between the maximum and minimum coefficient value is modified to embed a watermark bit. The watermark detection process is blind. In the watermark extraction processes if the difference between the maximum and minimum value coefficient is larger than the threshold, then the watermark bit is recovered as ‘1’otherwise ‘0’. The watermarking architecture was implemented on Xilinx Virtex IV technology and it operates on the 141.9 MHz frequency.

Sudip Ghosh et al.[22] proposed a fast reversible watermarking architecture based on contrast mapping. The watermark is embedded by replacing LSB of the image pixels or transformed pixels depending upon the control signals. The algorithm was implemented on Spartan 3E technology and device works at 45 MHz. Walsh-Hadamard transform based watermarking is proposed by Sudip Ghosh and others in [23]. The cover image and watermark transformed using FWHT. Transformed watermarked image is added to the coefficients of the cover image. The watermarking algorithm works binary as well as the grayscale watermark. The proposed algorithm operates on 259.202 MHz clock frequency.

Even the hardware architectures present the performance against the attacks is not discussed in the above work of literature. This literature discusses more the hardware implementation but the image quality and performance against the attack is equally important. Table I shows the comparison between the proposed watermarking algorithm and others.

No.	Proposed work	Type of watermark	Domain	Technology	Parallel Architecture	HIL test (Hardware in the Loop)	Performance against attack
1	[20]	Invisible	Spatial	FPGA	Y	N	N
2	[21]	Invisible	DWT Legal 5/3	FPGA	N	Y	N
3	[22]	Invisible	Spatial	FPGA	N	N	N
4	[29]	Invisible	FWHT	FPGA	N	N	N
5	**Proposed work**	**Invisible**	**DWT and DHT**	**FPGA**	**Y**	**Y**	**Y**

TABLE I. COMPARISON WITH EXISTING WATERMARKING ALGORITHM

3. DWT AND DHT

Most of the watermarking approaches are implemented with the single transform. There are many approaches implemented with DWT-DCT[30] and DWT-DCT-SVD[31]. In this paper, we have proposed a combined approach of DWT-DHT.

A Spatial localisation property of DWT made a promising candidate for digital watermarking. The Human Visual System (HVS) model is similar to the Multiresolution property of DWT transform. DWT can be implemented with lifting scheme which requires minimum computation power and hardware resources[32]. JPEG2000 is the newest version of one of the most popular image formats and it includes the DWT. Efficient VLSI implementations of DWT processors with watermarking unit became more important [36] [37]. CDF5/3 wavelet is used for lossless JPEG2000 standard. We have implemented CDF5/3 wavelet with the lifting scheme proposed in [38]. The lifting scheme calculates the DWT using spatial domain analysis, and consists of a series of Split, Predict and Update steps. The split step separates odd and even samples, and predict step predicts values in the odd set where -0.5 as the predict step coefficient. The Update step uses the new wavelet coefficients in the odd set to update the even set, where 0.25 as the update step coefficient. Figure 1 shows the lifting scheme for the CDF 5/3 wavelet.

Hadamard transform is the type of non-sinusoidal transform. The elements of the basis vector of the Hadamard transform take the only value of 1 and therefore it is faster and computationally efficient transform. This transform is well suited for the hardware implementations because it is only consisting of ±1 values. It has good energy compaction property. Hadamard transform is used in many signal processing applications such as image and video compression. Hadamard transform involves the shorter processing time as the processing involves only simple integer manipulation. So, computationally it is simpler to implement than any other orthogonal transforms[35]. DHT requires minimum hardware[33] compare to DCT[34].The proposed algorithm is implemented a fast DHT which less hardware and computing efficient. Figure 2shows the signal flow graph or the

fast Hadamard implementation. The 4×4 fast DHT transform requires four multipliers, two adders and two subtractions and the intermediate results are stored into memory.

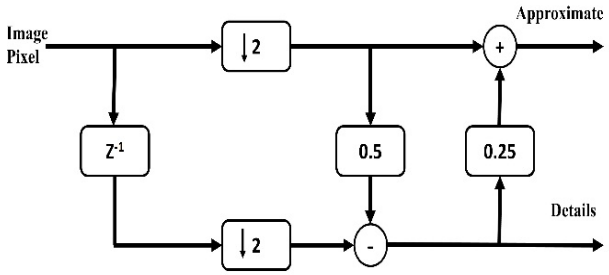


Figure 1. Lifting Scheme for 5/3 DWT

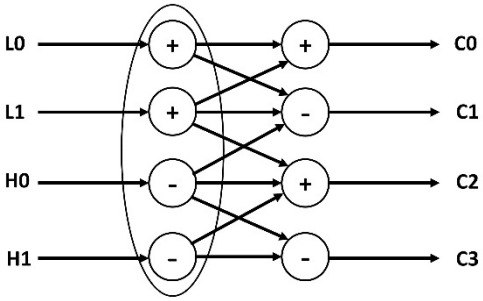


Figure 2. Signal flow graph of Fast DHT Transform

4. PROPOSED WATERMARKING ALGORITHM

For the watermarking process, the cover image is divided into blocks of equal size. In the first stage, the CDF5/3 wavelet is calculated separately for each block. This wavelet transformed coefficients are then rearranged in the specific order as shown in Figure 2. DHT is calculated for arranged wavelet block. Watermark is implemented in these transformed coefficients using the equation (1). The process is repeated till all blocks are over. There are many advantages of using block based watermarking. Multiple similar or different watermark can be embedded in blocks. Even if some of the watermarks are destroyed due to attack, still there is a chance to find the remaining watermark. As it is a repetitive structure, it can be easy to designed and implemented on hardware. It also requires less hardware. A finite state machine can be easily designed to control the flow. Figure 3 shows the algorithm for watermarking procedure.

$$IW_{coe,N}(x,y) = IM_{coe,N}(x,y) + G \times W(x,y) \qquad (1)$$

Where,
$IM_{coe,N}$:- Image coefficients of N^{th} block
$IW_{coe,N}$:- Watermarked Image coefficients of N^{th} block
G: - gain of the watermarking system
W: - watermark
x, y: - indices of the coefficients with the block

5. HARDWARE ARCHITECTURES FOR THE WATERMARKING ALGORITHM

In this section, we describe the VLSI architectures for the watermarking unit. Figure 4 shows the top module of the watermarking system. It consists of the memories for the image storage, address generator watermarking unit, CDF 5/3 wavelet filter, FDHT, inverse CDF5/3, and IDHT. Two separate memories are used to store the cover image and watermarked image. Address generator block generates a sequence of address in such a way that the block of pixels is extracted from the memory. Same addresses are used to store the watermarked image block with delay. This delay is equal to the time taken by the

watermarking unit to insert the watermark. Note that the block-processing unit is the most important unit. This unit calculates CDF 5/3 wavelets then DHT. The watermark is inserted using the equation (1). After inserting a watermark IDHT and IDWT are calculated. The address sequence from the address generator is used to store modified pixel values with delay. The hardware is designed for 64×64 and 8-bit grayscale image. This cover image is stored in a memory and the 4×4 blocks are generated using address generator unit. The detail sub-blocks of the watermarking algorithm illustrated bellow.

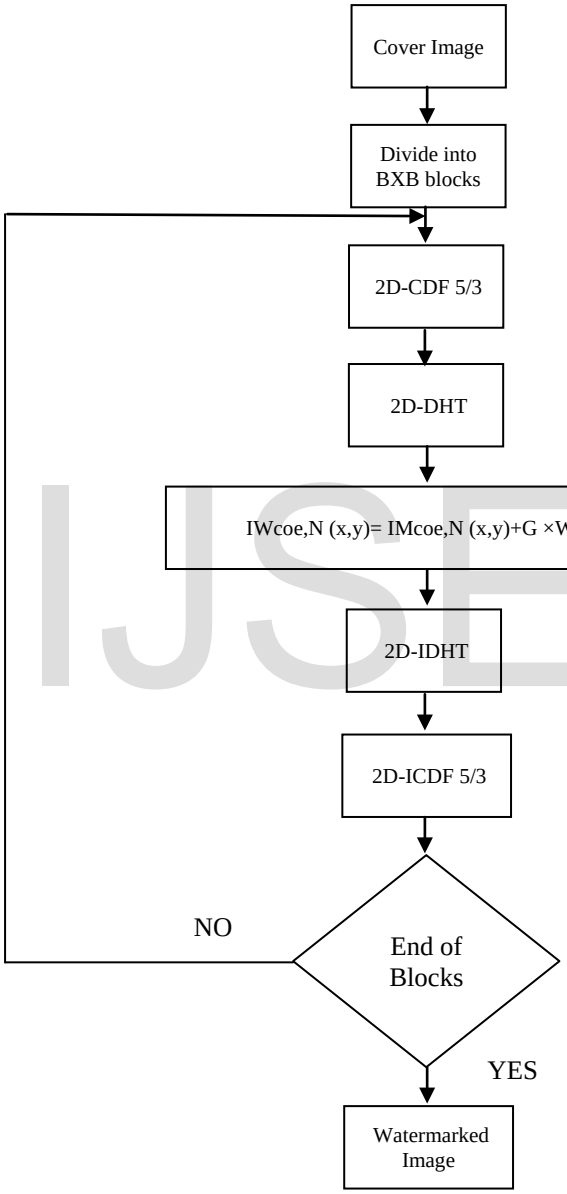


Figure 3. The perceptual flow of watermarking processes.

A. Address Generator

The address generator module generates the sequence of addresses to extract the block of pixels from the cover image. In the proposed algorithm, the block size is 4×4 and the cover image size is 64×64. Equation (4) is used to generate the address.

$$Address = (X + (Y \times 64) + (P \times 4) + (Q \times 256)) \quad (2)$$

Where X, Y, P and Q are the counters. Counter x counts from 0 to 3; each time when the counter finishes its count, counter Y is incremented by one. When one block pixels are over, P is incremented by one, and when 64 blocks are completed, Q is incremented by one. This process continues till all the 4096 blocks are over. Figure 5 shows the address generation mechanism. The addressing mechanism can be easily modified for the different block size and images for the different size.

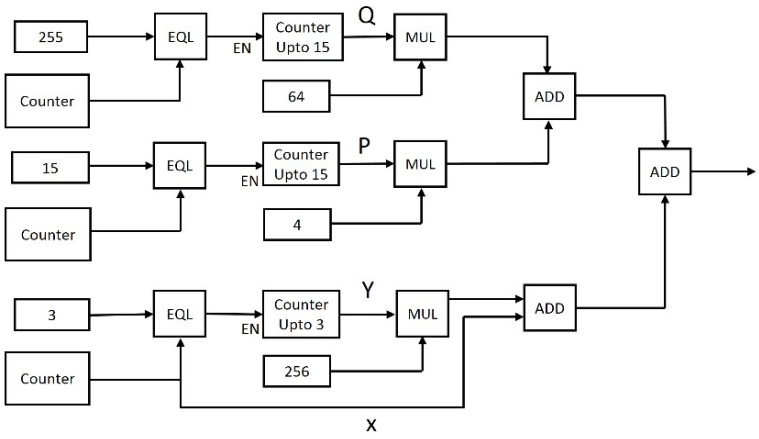


Figure 5. Address Generating Unit.

B. CDF 5/3 Wavelet Unit

The TDD (1:4) is used to divide the input data into even and odd samples. These samples are fed to the forward CDF 5/3 wavelet filter bank. A modified version of lifting scheme shown in figure 1 is implemented. This parallel implementation calculates all the coefficients at the same time. Serial and parallel combinations are used to optimise the area, power and speed. In the watermarking unit, image pixels are transformed into wavelet domain using CDF 5/3 wavelet filters. The first filter calculates the transform Column wise and the second filter calculates is row wise. Figure 5 and Figure 6 shows the implementation of forward CDF 5/3 filters and inverse CDF 5/3 wavelet respectively.

C. Discrete Hadamard Unit

The wavelet transformed coefficients are passed to the DHT block. DHT transform is the most suited transform for the hardware as it requires a minimum amount of hardware as compare to DCT. Figure 2 shows the signal flowgraph of the FDHT transform. Same is implemented in hardware. To calculate the 2D-DHT transform first it is applied row wise and column-wise. The same unit can be used to calculate its inverse. Figure 7 shows the hardware implementation of DHT.

D. Watermarking Unit

The DWT and DHT transformed values are then fed to the watermarking unit. The watermarking block then modifies these transformed values according to the equation (3). If the watermarking bit is one, then the gain value is added to the transformed pixel value, otherwise, zero is added. An equitation can be implemented using simple 2:1 multiplexer and added. If the watermarking bit is one then the value is added the pixel otherwise zero is added. Watermarking unit consists of four such units, each assigned

for one pixel value. This parallel implementation gives the higher speed for real-time applications. Figure 8 shows the watermarking unit.

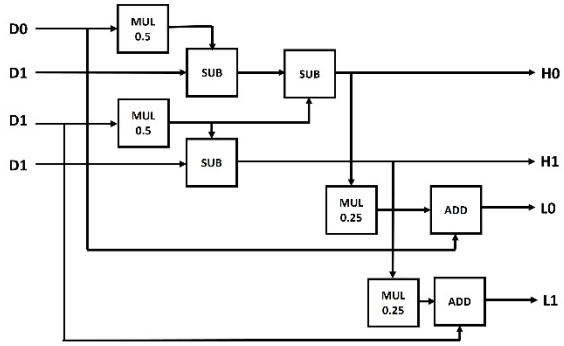


Figure 6. Forward CDF 5/3 Wavelet Unit.

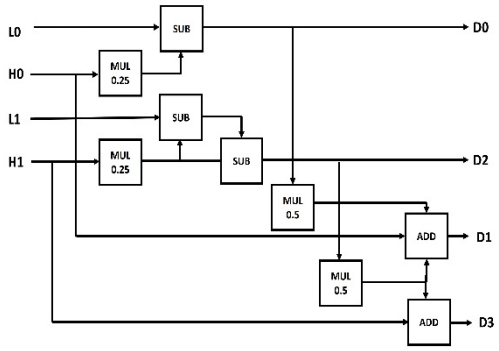


Figure 7. Inverse CDF 5/3 Wavelet Unit.

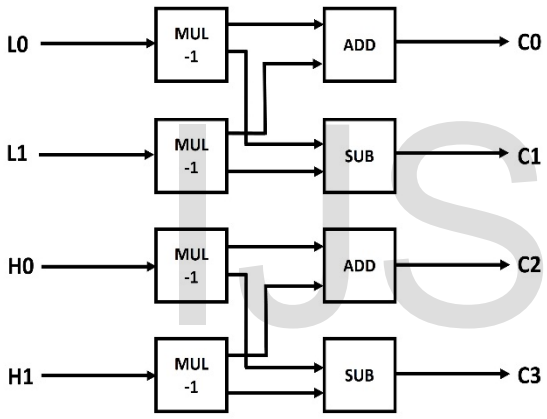


Figure 8. Forward DHT Unit.

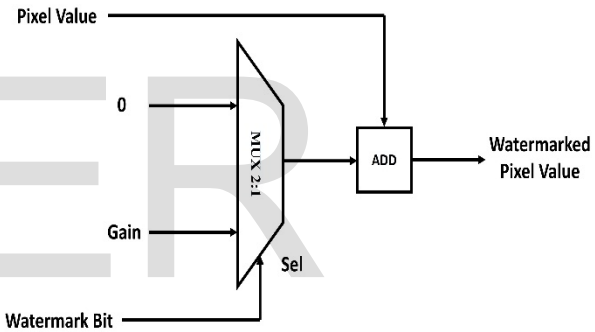


Figure 9. Watermarking Unit.

6. SYNTHESIS AND IMPLEMENTATIONS

The architecture was modelled using VHDL and functional simulation was performed using Xilinx ISE software. The algorithm was synthesised on XC3SD1800A-4FGG676C device from Xilinx Spartan-3A DSP technology. The hardware utilisation summary is presented in Table II. The Hardware in the Loop (HIL) technique was used to verify the results. The HIL was run at 33 MHz clock frequency. The power analysis is performed using Xilinx XPower analyser and it is estimated that the power consumption will be 138 mW. The design can operate with the maximum frequency of 49.985 MHz. Maximum path delay was estimated as 20.006 ns. Hardware utilisation shows the increase in a hardware requirement because of the parallel implementation of the filters. The comparison shows that the serial- parallel combination has reduced the IOBs. As the image is read serially into FPGA and read out as serially. As it can operate the device on the lower frequency to gives the real-time performance.

Logic Utilization	Utilization (%)				
	Proposed work	**[20]**	**[21]**	**[22]** (Encoder only)	**[29]**
Name of the Device	**XC3SD1800A-4FGG676C**	XC4VLX200-11FF1513	VIRTEX 4-VLXL00FF1148-2	SPARTAN 3E	XC7VX1140T-1FLG1930
Number of slice flip flops	**1**	12.15	0.12	-	-
Number of 4-input LUTs	**8**	-	0.26	27	-
Number of occupied slices	**11**	11.40	3.08	31	-
Number of bonded IOBs	**10**	42	13.12	46	-
Number of DSP48A	**3**	-	-	-	-
Operating Frequency(MHz)	**49.985**	364.31	141.9	45	259.2

TABLE II. COMPARISON OF HARDWARE UTILISATION

Figure 9. Original Image.

Figure 10. Watermarked Image and Watermark.

Figure 9 shows the Original image and figure 10 shows the watermarked image with the binary watermark. This shows the imperceptibility of the watermark.

7. WATERMARK DETECTION

The watermark can be detected using two methods blind and non-blind. The watermarking hardware can be fit into some electronic gadgets like camera, scanner etc. For watermark detection procedure, hardware implementation is not necessary. We have developed a software algorithm which detects the watermark by the blind as well as nonblind method.

In non-blind watermark detection, both original and watermarked image are necessary. The suspected image and original image both are divided into blocks. DWT and DHT are calculated for each block separately. These coefficients are subtracted using equation (3). If the subtracted magnitude is greater than τ, then it is detected as logic '1' else '0'.

$$W(x,y) = \begin{cases} 1 & \text{IWcoe, N (x, y)} - \text{IMcoe, N (x, y)} > \tau \\ 0 & otherwise \end{cases} \quad (3)$$

' τ ' represents threshold for non-blind detection

In blind watermark detection, the binary watermark is treated as PN sequence. The suspected image is divided into B×B blocks, and DWT and then DHT coefficients are calculated. The correlation between watermark and image coefficient block is calculated using equation (4)

$$\gamma = \frac{\sum_m \sum_n (IW_{coe,N}(x,y) - \overline{IW}_{coe,N})(W^*(x,y) - \overline{W^*})}{\sqrt{\sum_m \sum_n (IW_{coe,N}(x,y) - \overline{IW}_{coe,N}) \sum_m \sum_n (W^*(x,y) - \overline{W^*})}} \quad (4)$$

If γ> ρ, then the watermark is detected. Whereas 'ρ' is the threshold for blind detection.

8.Performance Against Attacks

In this section, we the discussed performance of the watermarking algorithm against various attacks by standard bench mark software. StirMark is one of the earliest benchmark software. The StirMark includes several attacks like compression, geometric transformation, noise addition etc. The geometric attacks include rotation, cropping, scaling and geometric transformation with medium compression. Some of the results are summarised in Table 3. As it can be seen from Table 3 that watermarking algorithm is well sustained to geometric and signal processing attacks.

Conclusion

In this paper, we proposed a novel watermarking algorithm based on CDF5/3 and DHT transform. We have developed the efficient hardware architecture which can be used with lossless JPEG2000 compression standard. From the device utilisation summary, it is clear that the algorithm requires less hardware. Serial and parallel architecture gives low power consumptions, minimum hardware utilisation and better speed. The experimental results showed that the proposed scheme of the watermarking scheme is imperceptible and robust against geometric attacks and signal processing attacks.

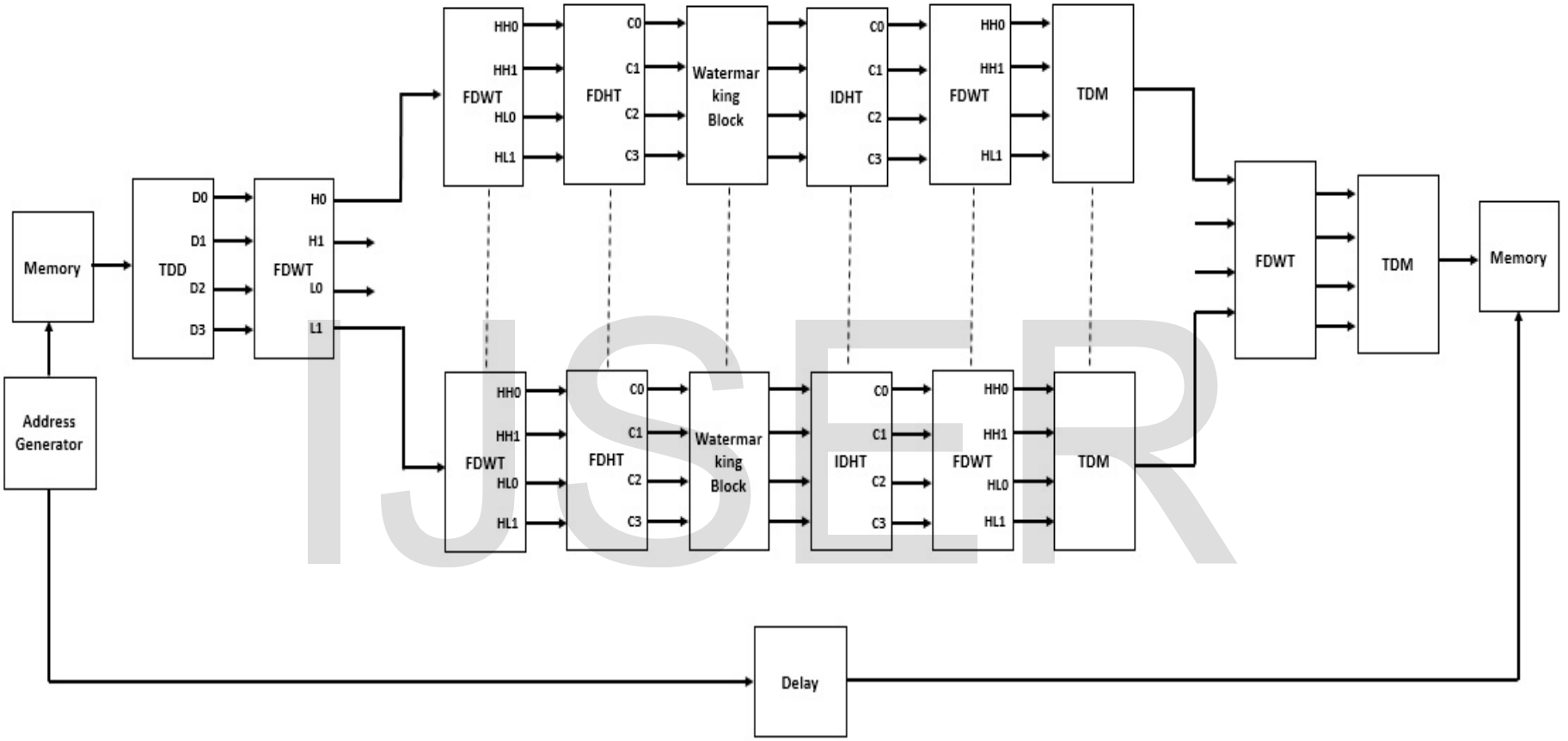


Figure 4. Hardware Architecture of watermarking algorithm.

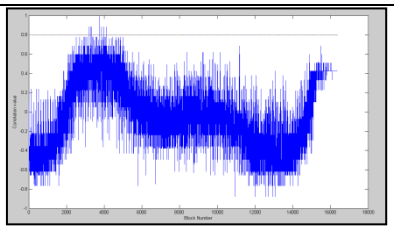
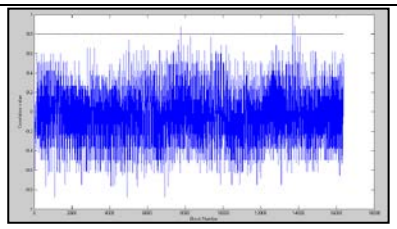
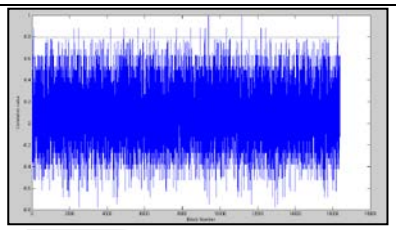
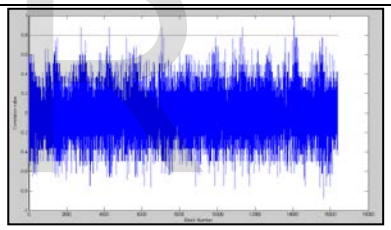
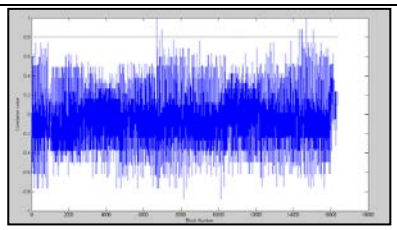
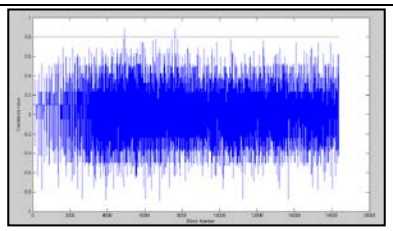
Attacked Image	Recovered Watermark	
	Non-Blind method	Blind method
 AFFINE_3		
 LATESTRNDDIST_1		
 NOISE_20		
 RESC_110		
 RML_60		
 ROT_15		

TABLE III. PERFORMANCE AGAINST VARIOUS ATTACKS AND BLIND, NON-BLIND DETECTION OF WATERMARK

References

[1] R. Chandramouli, N. Memon, and M. Rabbani, "Digital Watermarking," *Encycl. Imaging Sci. Technol.*, vol. 2939, no. 1, pp. 1–21, 2002.
[2] I. Cox, M. Miller, and J. Bloom, *Digital Watermarking and Steganography*. Elsevier, 2007.
[3] C. Neubauer, F. Siebenhaar, and K. Brandenburg, "Technical Aspects of Digital Rights Management Systems." Audio Engineering Society, p. 5688, 2002.
[4] A. Cheddad, J. Condell, K. Curran, and P. Mc Kevitt, "Digital image steganography: Survey and analysis of current methods," *Signal Processing*, vol. 90, no. 3, pp. 727–752, 2010.
[5] G. Coatrieux, L. Lecornu, B. Sankur, and C. Roux, "A review of image watermarking applications in healthcare," in *Annual International Conference of the IEEE Engineering in Medicine and Biology - Proceedings*, 2006, pp. 4691–4694.
[6] J. Seitz, *Digital Watermarking for Digital Media*. Information Science Pub, 2005.
[7] T. Bianchi and A. Piva, "Secure watermarking for multimedia content protection: A review of its benefits and open issues," *IEEE Signal Processing Magazine*, vol. 30, no. 2. pp. 87–96, Mar-2013.
[8] V. M. Potdar, S. Han, and E. Chang, "A survey of digital image watermarking techniques," in *INDIN '05. 2005 3rd IEEE International Conference on Industrial Informatics, 2005.*, 2005, pp. 709–716.
[9] C. Das, S. Panigrahi, V. K. Sharma, and K. K. Mahapatra, "A novel blind robust image watermarking in DCT domain using inter-block coefficient correlation," *AEU - Int. J. Electron. Commun.*, vol. 68, no. 3, pp. 244–253, 2014.
[10] P. Meerwald and A. Uhl, "Survey of wavelet-domain watermarking algorithms," *Proc. SPIE*, vol. 4314, pp. 505–516, Aug. 2001.
[11] H. Kasban, "A spiral based image watermarking scheme using Karhunen–Loeve and discrete hartley transforms," *Multidimens. Syst. Signal Process.*, pp. 1–23, Sep. 2015.
[12] S. Kumar and A. Dutta, "Performance analysis of spatial domain digital watermarking techniques," in *2016 International Conference on Information Communication and Embedded Systems (ICICES)*, 2016, pp. 1–4.
[13] S. Ghosh, B. Kundu, D. Datta, S. P. Maity, and H. Rahaman, "Design and implementation of fast FPGA based architecture for reversible watermarking," in *2013 International Conference on Electrical Information and Communication Technology (EICT)*, 2014, pp. 1–6.
[14] P. U. Lande and S. N. Talbar, "FPGA Implementation of Image Adaptive Watermarking Using Human Visual Model," *ICGST-PDCS J.*, vol. 9, no. 1, pp. 17–22, 2009.
[15] A. Akter, Nur-E-Tajnina, and M. A. Ullah, "Digital image watermarking based on DWT-DCT: Evaluate for a new embedding algorithm," in *2014 International Conference on Informatics, Electronics and Vision, ICIEV 2014*, 2014.
[16] H. T. Hu and L. Y. Hsu, "Collective blind image watermarking in DWT-DCT domain with adaptive embedding strength governed by quality metrics," *Multimedia Tools and Applications*, no. 1, pp. 1–20, 2016.
[17] A. Al-Haj, "Combined DWT-DCT Digital Image Watermarking," *J. Comput. Sci.*, vol. 3, no. 9, pp. 740–746, 2007.
[18] A. Kumar and S. Mayank, "Hybrid Technique for Robust and Imperceptible Image Watermarking in DWT – DCT – SVD Domain," *Natl. Acad. Sci. Lett.*, vol. 37, no. August, pp. 351–358, 2014.
[19] H. T. Hu, L. Y. Hsu, and J. Garcia-Alfaro, "Exploring DWT-SVD-DCT feature parameters for robust multiple watermarking against JPEG and JPEG2000 compression," *Comput. Electr. Eng.*, vol. 41, no. C, pp. 52–63, 2015.
[20] G. Akkad, M. Elhassan, and R. Ayoubi, "FPGA Hardware Architecture for Stereoscopic Image Compression Based on Block Matching, Watermarking and Hamming Code," in *Image Processing, Applications and Systems (IPAS), 2016 International*, 2016, pp. 1–5.
[21] J. Ahmed, A. Aziz, and P. Akhtar, "FPGA based efficient architecture for image watermarking using Wavelet Co-efficients Quantization," in *2014 International Conference on Open Source Systems & Technologies*, 2014, pp. 105–112.
[22] S. Ghosh, B. Kundu, D. Datta, S. P. Maity, and H. Rahaman, "Design and Implementation of Fast FPGA Based Architecture for Reversible Watermarking," in *2013 International Conference on Electrical Information and Communication Technology*, 2013, pp. 1–6.
[23] S. Ghosh, N. Das, S. Das, S. P. Maity, and H. Rahaman, "FPGA and SoC based VLSI architecture of reversible watermarking using rhombus interpolation by difference expansion," *11th IEEE India Conf. Emerg. Trends Innov. Technol. INDICON 2014*, 2015.
[24] S. P. Mohanty, N. Ranganathan, and R. K. Namballa, "VLSI implementation of invisible digital watermarking algorithms towards the development of a secure JPEG encoder," in *2003 IEEE*

Workshop on Signal Processing Systems (IEEE Cat. No.03TH8682), 2003, pp. 183–188.

[25] S. P. Mohanty, "A secure digital camera architecture for integrated real-time digital rights management," *J. Syst. Archit.*, vol. 55, no. 10–12, pp. 468–480, 2009.

[26] H. Hyun Lim, S.-Y. Soon-Young Park, S.-J. Seong-Jun Kang, and W.-H. Wan-Hyun Cho, "FPGA implementation of image watermarking algorithm for a digital camera," in *2003 IEEE Pacific Rim Conference on Communications Computers and Signal Processing (PACRIM 2003) (Cat. No.03CH37490)*, 2003, vol. 2, pp. 1000–1003.

[27] Y.-C. F. Y.-C. Fan, L.-D. V. L.-D. Van, C.-M. H. C.-M. Huang, and H.-W. T. H.-W. Tsao, "Hardware-efficient architecture design of adaptive visible watermarking," *Proc. Ninth Int. Symp. Consum. Electron. 2005. (ISCE 2005).*, no. Cic, pp. 399–403, 2005.

[28] G. N. S. Pankaj U.Lande, S.N. Talbar, "FPGA implementation of image adaptive watermarking using human visual model," *ICGST-PDCS*, vol. 9, no. 1, pp. 17–22, 2009.

[29] S. Ghosh, A. Biswas, S. P. Maity, and H. Rahaman, "Design of a Low Complexity and Fast Hardware Architecture for Digital Image Watermarking in FWHT Domain on FPGA," *2014 Fifth Int. Symp. Electron. Syst. Des.*, pp. 68–72, 2014.

[30] K. Deb, M. S. Al-Seraj, M. M. Hoque, and M. I. H. Sarkar, "Combined DWT-DCT based digital image watermarking technique for copyright protection," in *2012 7th International Conference on Electrical and Computer Engineering*, 2012, pp. 458–461.

[31] H. T. Hu, L. Y. Hsu, and J. Garcia-Alfaro, "Exploring DWT-SVD-DCT feature parameters for robust multiple watermarking against JPEG and JPEG2000 compression," *Comput. Electr. Eng.*, vol. 41, no. C, pp. 52–63, 2015.

[32] C. J. Lian, K. F. Chen, H. H. Chen, and L. G. Chen, "Lifting based discrete wavelet transform architecture for JPEG2000," in *ISCAS 2001 - 2001 IEEE International Symposium on Circuits and Systems, Conference Proceedings*, 2001, vol. 2, pp. 445–448.

[33] A.Amira, "An FPGA based parameterisable system for discrete Hartley transforms implementation," in *Proc. 2003 International Conference on Image Processing, ICIP 2003*, 2003, vol. 2, p. II-567-70.

[34] L. V. Agostini, I. S. Silva, and S. Bampi, "Pipelined fast 2D DCT architecture for JPEG image compression," in *Proceedings - 14th Symposium on Integrated Circuits and Systems Design, SBCCI 2001*, 2001, pp. 226–231.

[35] B. N. C. K.Veeraswamy , S. Srinivaskumar, "Designing Quantization Table for Hadamard Transform based on Human Visual System for Image Compression," *ICGST-GVIP J.*, vol. 7, no. 3, pp. 31–38, 2007.

[36] M. Nagabushanam, "Design and Implementation of Parallel and Pipelined Distributive Arithmetic Based Discrete Wavelet Transform IP Core," *Proc. Eur. …*, vol. 35, no. 3, pp. 378–392, 2009.

[37] K. Wagh, P. Dakhole, and G. Vinod, "Design & implementation of JPEG2000 encoder using VHDL," *Proc. world Congr. …*, vol. I, no. 1, 2008.

[38] Y. H. Seo and D. W. Kim, "VLSI architecture of line-based lifting wavelet transform for motion JPEG2000," *IEEE J. Solid-State Circuits*, vol. 42, no. 2, pp. 431–440, 2007.