# VLSI Implementation of Reversible Watermarking using RCM

Riddhi Patel
VLSI & Embedded System Design
Gujarat Technological University PG School
Ahmedabad, India
*Riddhi16211@gmail.com*

Mousami Turuk
Assistant Professor
PICT
Pune, India
*mpturuk@pict.edu*

***Abstract*** — In this paper the design and implementation of image watermarking algorithm called reversible contrast mapping (RCM) is given. Watermarking is to embed the one important information into the cover media to provide the security to data. Reversible watermarking is one of the important scheme of the watermarking which is useful when reconstruction of original image from watermarked data is necessary after using it like medical and military images. We have used reversible contrast mapping (RCM) algorithm for inserting watermark of size 64×64 in original image of size 128×128. It is integer transform applied on pixel pairs. It gives high embedding rate at lower mathematical complexity and it does not require data compression with lowest complexity.

***Keywords***- *Watermarking, RCM (Reversible Contrast Mapping)*

*****

## I. INTRODUCTION

Digital watermarking [1] is an efficient tool to prevent unauthenticated use of data. Digital watermarks may be used to verify the authenticity or integrity of the original data. Nowadays, it is prominently used for tracing copyright infringements and for banknote authentication. Digital watermarking is broadly classified depending on the type of signal like audio watermarking, image watermarking, video watermarking and database watermarking etc. The present work is focused on image watermarking.

In image watermarking, the digital information (like a digital image, a digital signature or a random sequence of binary numbers) is embedded into an image. The embedded information may or may not be perceptible after watermarking and therefore falls into the category of visible or invisible watermarking respectively. Depending on the robustness of the watermark, it can also be categorized as robust or fragile watermarking [2].

One limitation of watermarking-based authentication schemes is the distortion inflicted on the host media by the embedding process. Although the distortion is often insignificant, it may not be acceptable for some applications, especially in the areas of medical imaging and military applications. Therefore, watermarking scheme capable of removing the distortion and recovering the original media after passing the authentication is desirable. Schemes with this capability are often referred to as reversible watermarking schemes [3]. Recently reversible watermarking (RW) becomes popular in the areas of e-health care (telemedicine, telesurgury, remote patient monitoring), military and legal domain due to its high imperceptibility and secured communication. Various Reversible Watermarking techniques have been proposed with different type of algorithm [4][5]. Popular techniques of reversible watermarking are:

i) Difference Expansion
ii) Histogram bin Shifting
iii) Data hiding using Integer Wavelet Transform
iv) Contrast Mapping
v) Integer Discrete Cosine Transform

However, in this paper a reversible watermarking technique RCM is implemented using Xilinx. The choice of this technique includes its low computational complexity and robustness. The primary goal of the proposed design is to achieve high speed hardware efficient VLSI architecture.

The synopsis of the paper is as follows: In section II, we discussed our proposed design. In section III, we present the watermark embedding and extraction algorithm and flow chart. Section IV holds the simulated result using Xilinx ISE web pack. Finally, concluded in section V.

 ISSN: 2321-8169
2724 - 2727

## II. PROPOSED WORK

Let [0, L] be image gray level dynamic range (L is 255 for eight-bit per pixel image) and let (x, y) be a pair of pixels. In [3], the forward RCM transforms pair of pixels (x, y) into pairs of pixel (x', y') as follows:

$$X' = \frac{5X - Y}{4} \quad (1)$$

$$Y' = \frac{5Y - X}{4} \quad (2)$$

To prevent overflow and underflow, the transform is restricted to a subdomain D ⊂ [0, L]×[0, L] defined in (3) as follows:

$$0 \le \frac{5X-Y}{4} \le L \quad , \quad 0 \le \frac{5Y-X}{4} \le L \quad (3)$$

The inverse transform is defined as follows:

$$X = \frac{5X' + Y'}{6} \quad (4)$$

$$Y = \frac{5Y' + X'}{6} \quad (5)$$

Mathematically, the above transform may be interpreted as an intercept in straight line where a given (x, y) is stretched to (x', y') in the opposite sides as shown in given figure (1).

In other words, dynamic range of the transformed pair is extended with respect to the original pairs. In watermarking, the forward transform should not introduce much visual artifacts. By taking the sum and the difference of (1) and (2), one gets x' + y' = x + y and x' - y' = 3(x - y). This means that RCM preserves the gray level averages and increases the difference between the transformed pixels. Contrast manipulation is used to improve visual appearance in image processing with improved peak-signal-to-noise-ratio (PSNR) value.

## III. WATERMARKING ALGORITHM

In this section, the watermark embedding and extraction procedure using RCM is presented.

### A. *Watermark embedding Algorithm:*

1. Divide the cover image into several 2×1 segments (*X*, *Y*) called pixel pairs.
2. Define one proper threshold. If $|X - Y|/4 \ge \delta$, the image segment (*X*, *Y*) cannot be used to embed secret data.
3. If $|X - Y|/4 < \delta$. Apply transform (1) and (2) on pair of pixels.
   To prevent overflow and underflow, the transform is restricted to a subdomain $D \subset [0,255] \times [0,255]$ defined by the equation (3).
4. If $X' > 255$, $Y' > 255$, $X' < 0$ , $Y' < 0$, or $|X - Y|/4 \ge \delta$, record the LSB of *X* and then replace it as 0.
5. If $0 \le X', Y' \le 255$, classify if the pixel value of (*X*, *Y*) is an even or odd number and embed with the following methods:
   a) If it is composed of odd pixel values, set the LSB of *X* to "0," and embed the watermark information into the LSB of *Y*.
   b) If it is not composed of all odd pixel values, transform (x, y) into ($X'Y'$) and replace the LSB of $X'$ as 1. Lastly, embed the secret message into the LSB of $Y'$.

The dataflow graph of watermark embedding algorithm is shown in below figure.

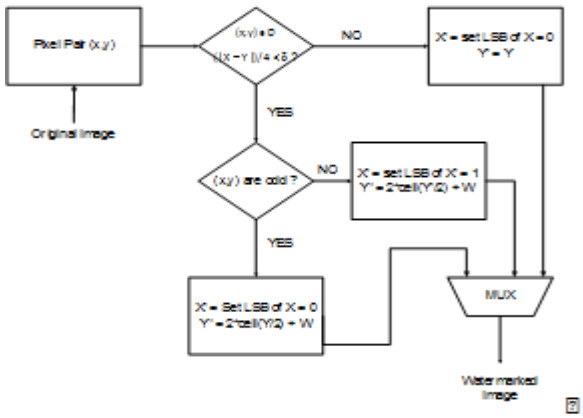


Figure 1:- Data flow of watermark embedding

When the receiver receives the watermarked image, it extracts the secret data and recovers the image by the following procedure:

### B. *Watermark extraction algorithm:*

1. Divide the received image into several 2×1 segments ($X''$, $Y''$).

 ISSN: 2321-8169
2724 - 2727

2. Classify if $X'$ is an even or odd number and extract the message with the following three strategies:

   a) If $X'$ is an odd number, extract the LSB of $Y'$ and that is the embedded message. In order to recover the image, we replace the LSB of $X'$ and $Y'$ as 0 separately and utilize and perform inverse transform given in equation (3) and (4) to recover the image.

   b) If $X'$ is an even number, replace the LSB of ($X'$, $Y'$) as 1 temporarily and carry into the Equation 1 for calculation. If there is no overflow or underflow problem, it means that there is embedded message in this segment. Then, we replace the LSB of $X'$ and $Y'$ as 1 to recover the cover image.

   c) If $X'$ is an even number and there is an overflow or underflow problem, it means that there is no embedded message in this segment. In order to restore the cover image, we utilize the previous recorded LSB of $X$ to replace the LSB of $X'$.

The dataflow graph of watermark extraction algorithm is shown in below figure 2.

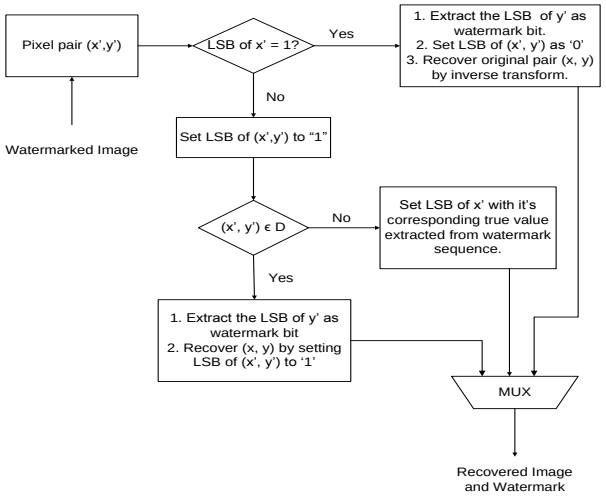


Figure 2:- data flow of watermark extraction

## IV. SIMULATION RESULT

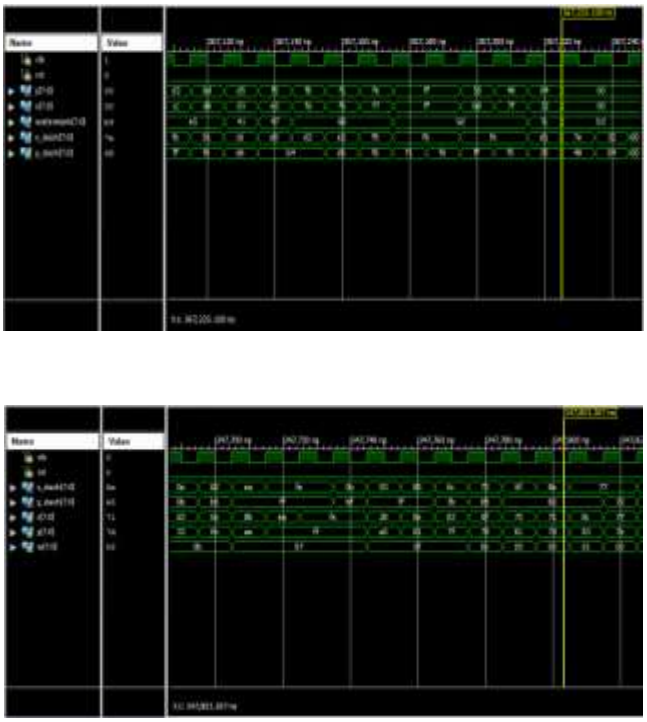

Figure 3. Simulation results of watermark embedding and extraction procedure

## V. CONCLUSION

This paper analyses approach to spatial domain reversible contrast mapping (RCM) image watermarking in detail. Also, it reviews the VLSI implementation of RCM. We have seen that it does not need additional data compression. In terms of mathematical complexity, it is reduced for both marking and decoding by using some low complexity bit manipulation. So RCM scheme is very appropriate for real-time applications. The RCM scheme provides robustness to cropping.

## ACKNOWLEDGMENT

I would like to express my deepest appreciation to all those who provided me the possibility to complete this paper. A special gratitude I give to my guide Mrs. Mousami Turuk and course coordinator Mr. Suresh Sikka whose contribution in stimulating suggestions and encouragement helped me to coordinate my seminar especially in writing this paper.

## REFERENCES

[1]. Fridrich, J., Goljan, M., & Du, R. (2001). Invertible authentication watermark for JPEG images. *Proceeding of the IEEE International Conferenceon Information Technology*, 223–227.

[2]. Cox, I., Miller, M., & Jeffrey, B. (2002). *Digital*

ISSN: 2321-8169
2724 - 2727

*watermarking: Principlesand practice*. Morgan Kaufmann.

[3]. Van Leest, A., Van der Veen, M., &Bruekers, F. (2003). Reversible image watermarking. *Proceedings of the IEEE International Conference onImageProcessing, II*, 731–734.

[4]. J. Tian. Wavelet-based reversible watermarking for authentication. In E. J. Delp III and P. W. Wong, editors, Security and Watermarking of Multimedia Contents volume 4675 of Proc. of SPIE, pages 679-690, Jan. 2002

[5]. *Coltuc*, D., Chassery, J.M.: *Very Fast Watermarking* by Reversible Contrast Mapping. IEEE Signal Processing Letters 14, 255–258 (2007).

[6]. H. Maity, S. P. Maity, C. Delpha, "A Modified RCM for Reversible watermarking With Fpga Implementation", EUVIP, pp. 100-105, 2013.

[7]. S.Ghosh, B.Kundu, D.Datta, S.P.Maity and H. Rahaman, "Design and Implementation of Fast FPGA Based Architecture for Reversible Watermarking", International Conference on Electrical Information and Communication Technology (EICT), 2013.

[8]. J. M. Barton, "Method and apparatus for embedding authentication information within digital data", U.S. Patent 5646 997, 1997.

[9]. C. D. Vleeschouwer, J. E. Delaigle, and B. Macq, "Circular interpretation of histogram for reversible watermarking", in Proceedings of the IEEE 4th Workshop on Multimedia Signal Processing, pp. 345350, France, Oct. 2001