Microelectronics Journal 42 (2011) 82–88

Contents lists available at ScienceDirect

Microelectronics Journal

journal homepage: www.elsevier.com/locate/mejo

FPGA and ASIC implementation of robust invisible binary image watermarking algorithm using connectivity preserving criteria

P. Karthigaikumar [a,*], K. Baskaran [b,1]

[a] Department of Electronics and Communication Engineering, Karunya University, Coimbatore, India
[b] Department of Computer Science and Engineering, Government College of Technology, Coimbatore, India

ARTICLE INFO

Article history:
Received 14 December 2009
Received in revised form
20 August 2010
Accepted 23 August 2010
Available online 17 September 2010

Keywords:
Watermarking
FPGA
ASIC
Multimedia
Low power VLSI

ABSTRACT

Digital watermarking is the process of hiding information into a digital signal to authenticate the contents of digital data. There are number of watermarking algorithm implemented in software and few in hardware. This paper discusses the implementation of robust invisible binary image watermarking algorithm in Field Programmable Gate Array (FPGA) and Application Specific Integrated Circuits (ASIC) using connectivity preserving criteria. The algorithm is processed in spatial domain. The algorithm is prototyped in (i) XILINX FPGA (ii) 130 nm ASIC. The algorithm is tested in Virtex-E (xcv50e-8-cs144) FPGA and implemented in an ASIC.

1. Introduction

In recent years, distribution of digital contents is one of the rapid up-coming fields owing to the latest progress in network and communication area, it is necessary to protect the data during transmission. Digital watermarking is a solution to the copyright [1] protection and authentication of data in the network. The data may be an image, audio, video, text or graphics [2,3]. In this paper binary image is given as input to the algorithm. The Watermark is included in the input signal to provide authentication and ownership to the transmitting signal. In general, any watermarking scheme consists of following parts, such as the watermark, the encoder (insertion algorithm) and the decoder and comparator (verification or extraction or detection algorithm) [3,4]. The insertion algorithm inserts the watermark into the object, whereas the verification algorithm authenticates the object, determining both the owner and the integrity of the object. The watermarks can be applied either in spatial or in frequency domain (FFT, DCT [11] or wavelet). Even though spatial domain watermarking is less robust [5–7], the spatial domain schemes have less computational overhead compared with frequency domain schemes. The digital watermarks can be divided into four different types such as visible [7–10], invisible [7], robust and invisible fragile [12].

Each of the above watermarking domains is equally important due to its unique applications. There are numerous watermarking algorithms designed based on software and their implementations are reported in the literature [13,14]. According to [28] only a few hardware schemes have been proposed. As proof of that, [27] provides list of the most watermarking hardware implementations available in the current literature.

Software implementation of watermarking algorithm has less useful when the image size and bit depth grow to high value. Hence hardware description languages are used to implement the media applications. Once the design has been programmed using VHDL and the desired performance is achieved, it can be downloaded into an FPGA. Digital revolution in video applications has brought profound challenges in media security and encryption. Many techniques like including ownership protection, authentication, access control and annotation have been proposed to counter this challenge. Data hiding [26,29] is one of the technique to hide the secret data. It provides imperceptibility and robustness and has the ability to hide many bits.

In the proposed watermarking technique, the pixels are flipped and hidden from the intruders. Reconfigurable techniques [8,30] can be applied to different blocks of processing element so that different image blocks can be watermarked in parallel.

2. Related research

This section discusses the few hardware implementation of watermarking algorithms reported in the literature. The hardware

[*] Corresponding author. Tel.: +91 94862 60288.
E-mail addresses: Karthi_kumar_p@rediffmail.com (P. Karthigaikumar), baski_101@yahoo.com (K. Baskaran).
[1] Tel.: +91 9443661901

0026-2692/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.mejo.2010.08.023

may be FPGA or ASIC. The algorithm is implemented in different domain like spatial, DCT and wavelet.

Mathai et al.[16] proposed invisible watermarking algorithm in wavelet domain. The algorithm is implemented in 0.18 μm technology.

Tsai and Lu [17] proposed invisible watermarking algorithm in DCT domain. It inserts pseudorandom sequence [19–21] into the incoming signal. The algorithm is implemented in 0.25 μm technology and has a size of $3.064\times3.064\,\mathrm{mm}^2$. It consumes 62.78 mW power when operated at 50 MHz frequency and 3.3 V.

Garimella et al. [18] proposed invisible fragile watermarking algorithm in spatial domain. Here, after the decryption, the original and extracted watermark is compared for authentication. The algorithm is implemented in ASIC using 0.13 μm technology and has a size of $3453\times3453\,\mu\mathrm{m}^2$. It consumes 37.6 μW power when operated at 100 MHz frequency and 1.2 V.

Mohanty et al. [15]proposed a watermarking algorithm which uses two visible watermark in spatial domain. The algorithm is implemented in 0.35 μm technology and has size of $3.34\times2.89\,\mathrm{mm}^2$. It consumes 6.9286 mW power when operated at 292.27 MHz frequency and 3.3 V.

Mohanty et al. [24] proposed invisible robust and visible watermarking algorithm in DCT domain. The algorithm is implemented in 0.25 μm technology and it consumes 0.3 mW power when operated at two different frequency (70 and 280 MHz) and two different voltages (1.5 and 2.5 V).

Mohanty et al. [25] proposed invisible robust and invisible fragile watermarking algorithm in spatial domain. The algorithm is implemented in 0.35 μm technology and has a size of $15.01\times14.225\,\mathrm{mm}^2$. It consumes 2.0547 mW power when operated at 545 MHz frequency and 3.3 V.

It can be figured out from the above analysis that there is lot of room for contribution in this area. This paper focus on implementation of an invisible robust watermarking algorithm in FPGA and ASIC. The proposed watermarking technique wherein the pixels are flipped and hidden from the intruders.

3. Design of watermarking algorithm at the chip level

3.1. Invisible watermarking algorithm

This section explains an invisible watermarking algorithm [22,23] which is implemented in FPGA and ASIC. The algorithm with modifications can improve the performance and security which can be used for real-time watermarking.

An image signal is taken as the input and converted to decimal format using MATLAB. This process would give an array of 256 × 256 pixels. Each pixel value ranging from 0 to 255 levels of its intensity. Then it is converted into binary value using threshold value. The value '1' is used to represent black pixel and '0' is used for white pixel. The results are seen in Fig. 1.

Hence an image is converted into binary form and partitioned into 3 × 3 windows shown in Fig. 2.

The eight neighbors of the centre pixels (p_c) in a 3 × 3 neighborhood are $w1,w2,w3,w4,w5,w6,w7,w8$. Four formulae (Eqs. 1–4) are applied to each window to check the flippability of the centre pixel. One control unit operates black transition formula, another one white transition formula the other one Interior Right (IR) angle formula and the another one 'C' transitions. The pixel can be flipped so that an intruder cannot get the original bit. The flippability of a pixel depends on the transitions of the pixel to its 8 neighbors in 3 × 3 blocks, in particular the 4 connectivity and the 8 connectivity among pixels.

The number of uniform white transitions and the number of black transitions along the vertical and the horizontal directions are collectively defined as VH Transitions

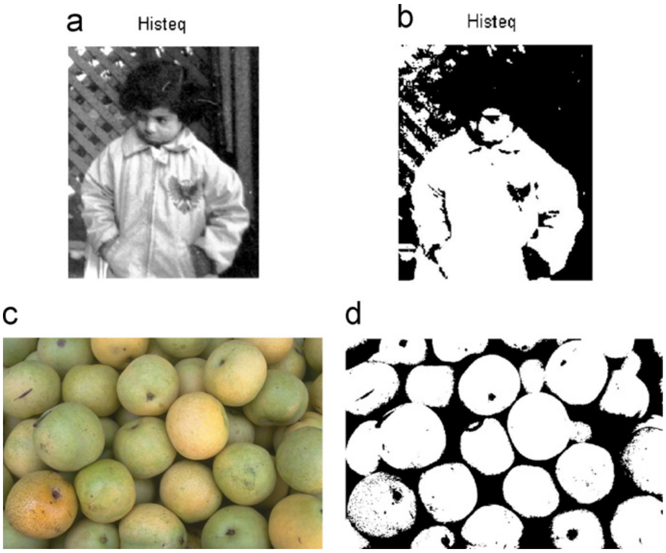

Fig. 1. MATLAB simulation results. (a) Gray scale input image (b) Output binary image of (a). (c) Input color image (d) Output binary image of (c).

W6	W7	W8
W5	Pc	W1
W4	W3	W2

Fig. 2. A 3 × 3 binary window of an image, whose centre pixel is to be watermarked [1].

$$\text{White transition } N_{\mathrm{vw}}=\sum_{i=1}^{3}(\sim p_{\mathrm{c}})(\sim w_{\mathrm{i}})(\sim w_{\mathrm{i}+4}) \qquad (1)$$

$$\text{Black transition } N_{\mathrm{vb}}=\sum_{i=1}^{3}(p_{\mathrm{c}})(w_{\mathrm{i}})(w_{\mathrm{i}+4}) \qquad (2)$$

The number of interior angle (IR) transitions in a 3 × 3 block is

$$\text{Interior angle transitions } N_{\mathrm{ir}}=\sum_{i=1}^{3}(\sim p_{\mathrm{c}})(\sim w_{2\mathrm{i}})(\sim w_{2\mathrm{i}-1})(\sim w_{2\mathrm{i}+1}) \qquad (3)$$

The number of 'C' transitions in a 3 × 3 block is

$$\text{Center pixel transitions } N_{\mathrm{c}}=\sum_{i=1}^{3}(p_{\mathrm{c}})(w_{2\mathrm{i}})(w_{2\mathrm{i}+1})(w_{2\mathrm{i}+2})(w_{2\mathrm{i}+3})(w_{2\mathrm{i}+4})$$

$$\cdot=\text{Logical “and”},\ \sim\ =\text{Logical “not”}$$
$$p_{\mathrm{c}}=\text{Centre pixel},\ w_{\mathrm{i}}=\text{Neighboring pixels} \qquad (4)$$

The flippability criterion is defined such that the centre pixel in a 3 × 3 block can be flipped if 'VH' transitions and 'IR' transitions and the 'C' transitions remain the same before and after flipping the pixel, which implies that flipping the pixel would not destroy the connectivity between the pixels and create extra clusters as well. These conditions are collectively known as the "Connectivity