Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 3, Issue. 4, April 2014, pg.1321 – 1327

RESEARCH ARTICLE

FPGA IMPLEMENTATION OF DIGITAL WATERMARKING SYSTEM

K.Tamilvanan[1], R.B.Selvakumar[2]

[1]PG Scholar, Electrical and Electronics Engineering & Anna University, India

[2]Assistant Professor, Electrical and Electronics Engineering & Anna University, India

[1] tamilvananmurthy@gmail.com, [2] rbsk.snsce@gmail.com

Abstract— Digital watermarking is an effective way to protect copyright of multimedia data even after its transmission. Current trends support digital image files as the cover file to hide another digital file with secret message or data. This paper proposes Discrete Wavelet Transform (DWT) to authenticate the multimedia image and it can convert the image from spatial domain to frequency domain. The idea behind the LSB algorithm is to insert the bits of the hidden message into the least significant bits of the pixels. Field Programmable Gate Array (FPGA) technology has become a viable target for the implementation of real time algorithms suited to video image processing applications. This System is developed on Xilinx Spartan3 Field Programmable Gate Array (FPGA) device using embedded development kit (EDK) tools from Xilinx. The results showed that the proposed algorithm has a very good hidden invisibility, good security and robustness for a lot of hidden attacks.

Key points: Digital watermarking, FPGA, LSB algorithm, DWT, EDK

I. INTRODUCTION

Digital watermarking is the process of embedding a message, called the payload, within the content of a host message, where the host message can be audio, still images, or video. Watermarking systems for audio and still imagery are often implemented in software due to the low data rate of these signals. For video streams, however, real-time watermarking is too expensive to implement in software, and hence there is a strong motivation for hardware implementation [1]. Watermarking is the art of invisible communication by concealing information inside other information.

The term watermarking is derived from the Greek and literally means “covered writing” [12]. A watermarking system consists of three elements: cover-object (which hides the secret message), the secret message and the secret-object (which is the cover object with message embedded inside it.) Given the proliferation of digital images on the internet, and the large redundant bits present in the digital representation of an image, images are the most popular cover objects for water marking [3].

© 2014, IJCSMC All Rights Reserved

A digital image is described using a 2-D matrix of the color intestines at each grid point (i.e. pixel) [5]. Typically, gray images use 8 bits, whereas colored utilizes 24 bits to describe the color model, such as RGB Model. The watermarking system which uses an image as the cover object is referred to as an image watermarking system [8]. There are several techniques to conceal information inside cover-image. The spatial domain techniques manipulate the cover-image pixel bit values to embed the secret information. The secret bits are written directly to the cover image pixel bytes. Consequently, the spatial domain techniques are simple and easy to implement [10]. The Least Significant Bit (LSB) is one of the main techniques in spatial domain image watermarking. The transform domain techniques embed the message in the frequency domain of the cover image. Typically, spatial domain techniques are easily detectable and have larger capacity. On the other hand, frequency-based watermarking has higher peak signal-to-noise ratio (PSNR) and is more secure. In this paper, the video file has been converted as a several frames [11]. By choosing a one image from that frames we have to hide a secret image. From that the sender can send the watermarked image to the receiver. In receiver side they have to extract the images from the input watermarked images [12].

II. LSB ALGORITHM

One of the most common techniques used in data hiding today is called least significant bit (LSB) insertion. This method is exactly what it sounds like; the least significant bits of the cover-image are altered so that they form the embedded information.

The LSB based image watermarking embeds the secret in the least significant bits of pixel values of the cover image (CVR). To illustrate LSB technique, we provide the following example. Suppose the CVR has the following two pixel values:

(0000 1010 0011 0010 0111 0100)
(1111 0101 1100 0011 1100 0111)

Also, assume that the secret bits are: 1011012. After embedding the secret bits, the result pixel values are:

(0000 101**1** 0011 001**0** 0111 010**1**)
(1111 010**1** 1100 001**0** 1100 011**1**)

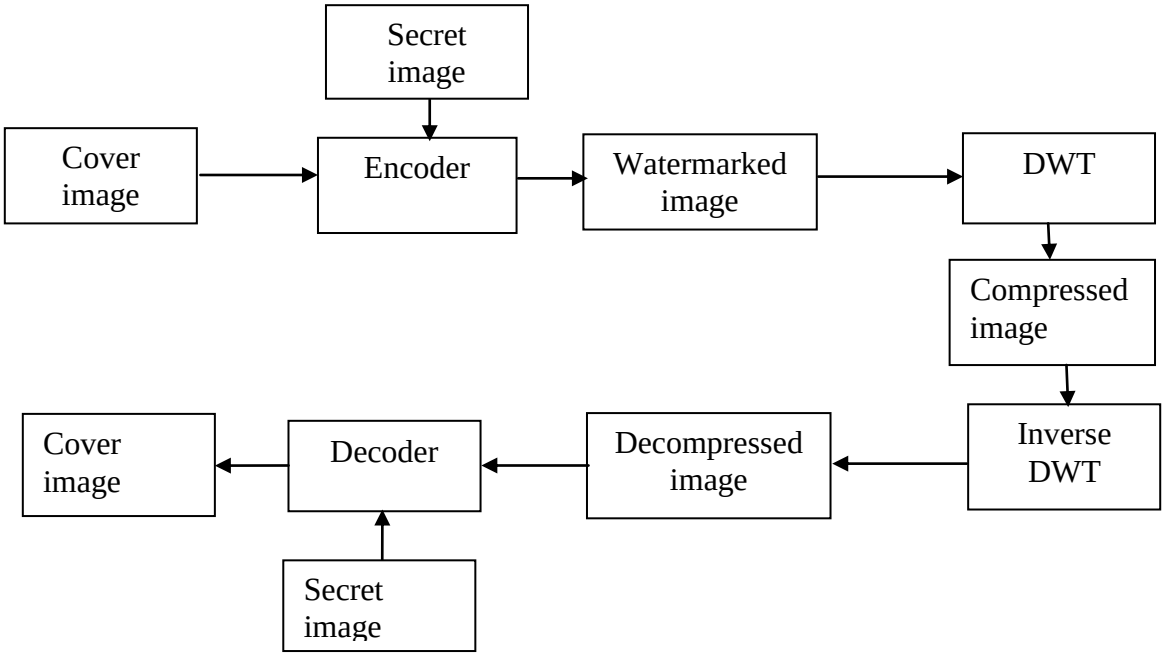

Fig. 1 Block diagram for watermarking by using LSB

The underlined bits indicate that the bits were changed from their original value. Only three bits in the cover image were modified. On average about half of the bits in the cover image will be modified when embedding

© 2014, IJCSMC All Rights Reserved

the secret image. The above LSB method limits the size of the secret data to eighth of the size of the CVR. LSB watermarking can be extended to embed secret information in the least n-bits to increase the capacity of the secret information n/8 the size of the CVR.

However, increasing n distorts secret-image. To illustrate the impact of the value of n on the secret-image, we performed several experimental runs on the test image, shown in Figure 1. In each run, we embed random data in the n least significant bits, where $1 \leq n \leq 7$. However, we need to introduce the methods to measure the quality and distortion in images [2]. Since the 8-bit letter A only requires eight bytes to hide it in, the ninth byte of the three pixels can be used to begin hiding the next character of the hidden message. A slight variation of this technique allows for embedding the message in two or more of the least significant bits per byte. This increases the hidden information capacity of the cover-object, but the cover-object is degraded more, and therefore it is more detectable. Other variations on this technique include ensuring that statistical changes in the image do not occur. Some intelligent software also checks for areas that are made up of one solid color. Changes in these pixels are then avoided because slight changes would cause noticeable variations in the area image processing. While LSB insertion is easy to implement, it is also easily attacked Slight modifications in the color palette and simple Image manipulations will destroy the entire hidden message.

Some examples of these simple image manipulations include image resizing and cropping. LSB approaches typically achieve high capacity. When the hidden message contains fewer bits than the cover image has pixels, we assume that the modifications are spread randomly around the cover image according to a secret key shared with the intended recipient of the secret image. This sort of data hiding is only suitable for images stored in bitmap form or lossless compressed. One should clearly distinguish this method (perhaps best called LSB replacement) from an alternative described, where the cover pixel values are randomly incremented or decremented so that the least significant bits match the hidden message (this should perhaps be called *LSB* matching).

In the latter case the message is still conveyed using the LSBs of the pixel values of the image, but the simple alteration to the embedding algorithm makes it much harder to detect. None of the methods discussed here will detect this alternative form of data hiding, and indeed it is a much more difficult task to do so: a detector for LSB matching in full color bitmaps is describe but it is ineffective for grayscale covers; another detector which works for full color images is also described. But it is only reliable for very large embedded messages and barely effective for grayscale covers. LSB replacement is by no means the best or even a sensible steganographic method. However we consider it extremely worthy of study because of its widespread use. A large majority of freely available data hiding software makes use of LSB replacement, but there is a more important reason: it can be performed without any special tools at all.

III. HARDWARE IMPLEMENTATION

In this proposed system VLSI architecture is designed and implemented, which is to perform the binary image processing with high speed and reduced complexity. For that, the coprocessor Micro blaze is converted into morphological processing architecture using Xilinx platform studio in system C language and then tested in Spartan 3EDK FPGA kit. RS232 cable is used for interfacing the test circuit with PC. This hardware implementation can overcome the shortages of previous works not only that it can achieve accuracy, noise and also the speed in computation and low power consumption also.

A. Processor design technique

The Micro Blaze embedded soft core is a reduced instruction set computer (RISC) optimized for implementation in Xilinx field programmable gate arrays (FPGAs). Field-programmable gate arrays (FPGA'S) are flexible and reusable high-density circuits that can be easily re-configured by the designer, enabling the VLSI design / validation /simulation cycle to be performed more quickly and less expensive. Increasing device densities have prompted FPGA manufacturers, such as Xilinx and Altera, to incorporate larger embedded components, including multipliers, DSP blocks and even embedded processors. One of the recent architectural enhancements in the Xilinx Spartan, Virtex family architectures is the introduction of the Micro Blaze (Soft IP) and PowerPC405 hard-core embedded processor. The Micro blaze processor is a 32-bit Harvard Reduced

Instruction Set Computer (RISC) architecture optimized for implementation in Xilinx FPGAs with separate 32-bit instruction and data buses running at full speed to execute programs and access data from both on-chip and external memory at the same time. An interrupt controller is available for use with the Xilinx Embedded Development Kit (EDK) software tools.

© 2014, IJCSMC All Rights Reserved

The processor will only react to interrupts if the Interrupt Enable (IE) bit in the Machine Status Register (MSR) is set to 1. On an interrupt the instruction in the execution stage will complete, while the instruction in the decode stage is replaced by a branch to the interrupt vector (address Ox 10). The interrupt return address (the PC associated with the instruction in the decode stage at the time of the interrupt) is automatically loaded into general purpose register. In addition, the processor also disables future interrupts by clearing the IE bit in the MSR. The IE bit is automatically set again when executing the RTID instruction.

Due to the advancement in the fabrication technology and the increase in the density of logic blocks on FPGA, the use of FPGA is not limited to anymore to debugging and prototyping digital circuits. Due to enormous parallelism achievable on FPGA and the increasing density of logic blocks, it is being used now as a replacement to ASIC solutions in a few applications. Soft cores are technology independent and require only simulation and timing verification after synthesized to a target technology.

B. Xilinx platform studio

The Xilinx Platform Studio (XPS) is the development environment or GUI used for designing the hardware portion of your embedded processor system. B. Embedded Development Kit Xilinx Embedded Development Kit (EDK) is an integrated software tool suite for developing embedded systems with Xilinx Micro Blaze and PowerPC CPUs. EDK includes a variety of tools and applications to assist the designer to develop an embedded system right from the hardware creation to final implementation of the system on an FPGA. System design consists of the creation of the hardware and software components of the embedded processor system and the creation of a verification component is optional. A typical embedded system design project involves: hardware platform creation, hardware platform verification (simulation), software platform creation, software application creation, and software verification. Base System Builder is the wizard that is used to automatically generate a hardware platform according to the user specifications that is defined by the MHS (Microprocessor Hardware Specification) file. The MHS file defines the system architecture, peripherals and embedded processors]. The Platform Generation tool creates the hardware platform using the MHS file as input. The creation of the verification platform is optional and is based on the hardware platform.

IV. RESULTS AND DISCUSSION

In this paper first going to convert the input video into number of image frames and then creates a header file for the input image by using MATLAB. In MATLAB we are creating the header files using GUI window. By using that header files as supporting file, fusing the two images by using LSB technique. Fig. 2 shows the GUI window to create header file for the input image and hidden image in MATLAB.

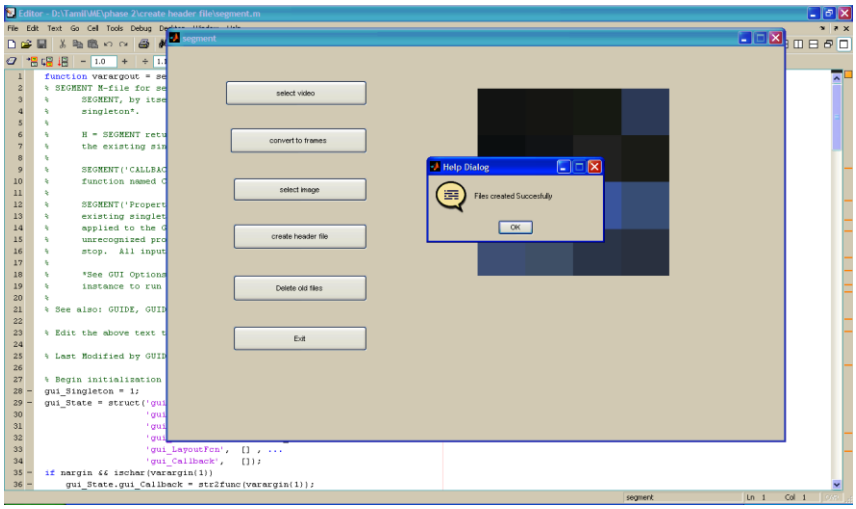

Fig. 2 GUI window to create header file

After convert the header file in the Mat lab the XPS could be processed. In XPS the impulse C language is used. The processing of the input image and the watermarked image could be processed in the SPARTAN 3EDK. These could be processed by converting these codings into the Bit streams. Then the net list will be created for that bit streams to get the proper output.

© 2014, IJCSMC All Rights Reserved

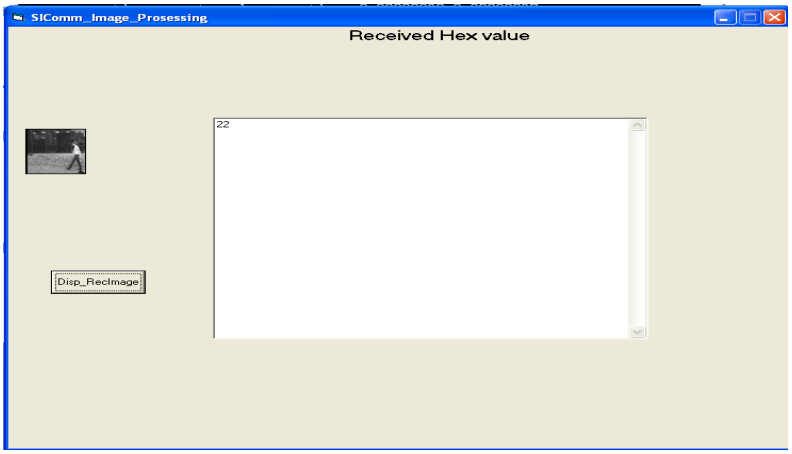

Fig. 3 Input image

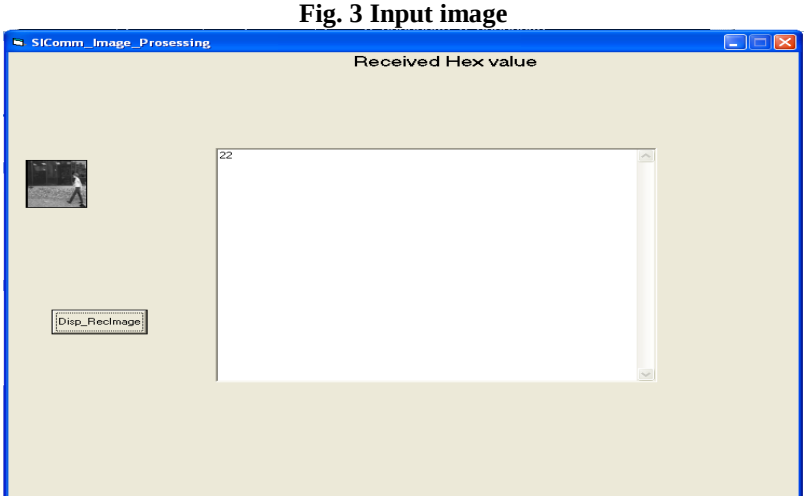

Fig. 4 Watermarked image

In this paper a GUI window can be created by using Visual Basic to shown the output obtained from Xilinx Spartan3 EDK. Fig. 3 shows the Input image and Fig. 4 shows the Watermarked image after LSB encryption obtained from Xilinx Spartan3 EDK.

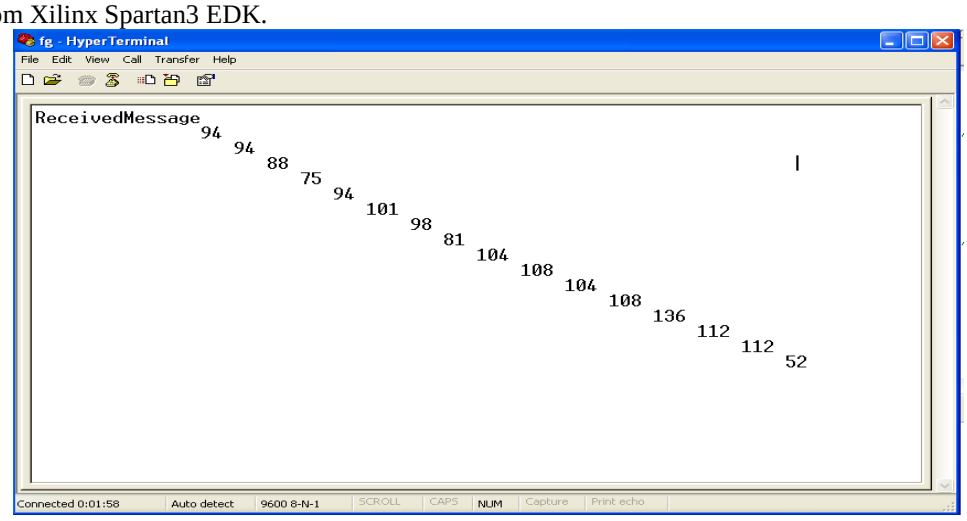

Fig. 5 Retrieved Image

Fig.5 shows the Original data after LSB decryption by using hyperterminal.

© 2014, IJCSMC All Rights Reserved

V. CONCLUSION

Here invisible watermarking technique is implemented by using LSB algorithm on FPGA and the outputs are verified by applying discrete wavelet transform technique to the secret image to get the better results. The hardware implementation of this digital watermarking could be significant in the copyright networks for the processing of the secret network based images. The related work for this implementation could be recognized over the processing of this LSB technique for the tamper proofing also. The hardware architecture resources also could be based on the bench mark system. This type of work using EDK can be extended to other applications of embedded system. In future, we are going to implement this video watermarking to ALTERA and VIRTEX to get the high speed of operation.

REFERENCES

[1] Sonjoy Deb Roy, Xin Li, Yonatan Shoshan, Alexander Fish, "Hardware Implementation of a Digital Watermarking System for Video Authentication", IEEE transactions on circuits and systems for video technology, vol. 23, no. 2, February 2013

[2] Vidyasagar M. Potdar, Song Han, Elizabeth Chang, "A Survey of Digital Image Watermarking Techniques", 3rd IEEE International Conference on Industrial Informatics (INDIN),2005, pp.709-716.

[3] Alessandro Piva , Franco Bartolini and Mauro Barni, "Managing Copyright in Open Networks",IEEE internet computing june 2002.

[4] Yonatan Shoshan, Alexander Fish, Xin Li, Graham Jullien, Orly Yadid-Pecht "Vlsi Watermark Implementations And Applications", International Journal "Information Technologies and Knowledge" Vol.2 / 2008

[5] Xin Li, Yonatan Shoshan, Alexander Fish, Graham Jullien, Orly Yadid-Pecht, "Hardware Implementations Of Video Watermarking", International Book Series "Information Science and Computing" 2008.

[6] Ingemar J. Cox, Joe Kilian, F. Thomson Leighton, and Talal Shamoon, "Secure Spread Spectrum Watermarking for Multimedia", IEEE Transactions On Image Processing, Vol. 6, No. 12, December 1997.

[7] Franco Bartolini, Anastasios Tefas and Mauro Barni, "Image Authentication Techniques for Surveillance Applications", IEEE Transactions on Image Processing,vol. 3,No. 8,August 2002.

[8] Jana Dittmanna, Martin Steinebacha, Ivica Rimacb, Stephan Fischerb, and Ralf Steinmetza,b, "Combined video and audio watermarking: Embedding contentinformation in multimedia data", International Journal of research and Technology, march 2009.

[9] A. D. Gwenael and J. L. Dugelay, "A guide tour of video watermarking," *Signal Process. Image Commun.*, vol. 18, no. 4, pp. 263–282, Apr. 2003.

[10] X. Li, Y. Shoshan, A. Fish, G. A. Jullien, and O. Yadid-Pecht, "Hardware implementations of video watermarking," in *International Book Series on Information Science and Computing*, no. 5. Sofia, Bulgaria: Inst. Inform. Theories Applicat. FOI ITHEA, Jun. 2008, pp. 9–16 (supplement to the *Int. J. Inform. Technol. Knowledge*, vol. 2, 2008).

[11] S. P. Mohanty. (1999). *Digital Watermarking: A Tutorial Review* [Online]. Available: http://www.linkpdf.com/download/dl/ digital-watermarking-a-tutorial-review-.pdf [8] F. Hartung and B. Girod, "Watermarking of uncompressed and compressed video," *IEEE Trans. Signal Process.*, vol. 66, no. 3, pp. 283–302, May 1998.

[12] F. Hartung and B. Girod, "Watermarking of uncompressed and compressed video," *IEEE Trans. Signal Process.*, vol. 66, no. 3, pp. 283–302, May 1998.

© 2014, IJCSMC All Rights Reserved

AUTHORS' BIOGRAPHY

K. Tamilvanan received the B.E in Electronics and Communication Engineering from Karpagam College of Engineering, Coimbatore, in 2012, and currently pursuing M.E degree in Applied Electronics in SNS College of Engineering, Coimbatore. His area of interest includes Digital Image Processing, VLSI Design, communication system and Digital Electronics.

R.B.Selvakumar received the Diploma in Electrical and Electronics Engineering from PSG Polytechnic College, Coimbatore, in 2002, and the B.E. degree in Electrical and Electronics Engineering in 2009 from Coimbatore Institute of Technology, Coimbatore under Anna University. He received M.E. degree in Power Electronics and Drives in 2013 under Anna University. He is working as Assistant Professor in SNS College of Engineering, Coimbatore. His research area of interest includes Electrical Machines, Control Systems, Power Electronics Applications to Renewable Energy Systems, Power Quality.

© 2014, IJCSMC All Rights Reserved