Grenze International Journal of Engineering and Technology, January Issue

Configurable Multi-Image Transposed Steganography on FPGA

Abhay Chopde[1], Om Tidke[2], Sahish Pandav[3] and Madhura Pande[4]
[1-4]Department of Electronics and Telecommunication Engineering, Vishwakarma Institute of Technology, Pune, India
Email: Abhay.chopde@vit.edu, baban.om21@vit.edu, sahish.pandav21@vit.edu, madhura.pande211@vit.edu

***Abstract*—The paper investigates the application of image steganography techniques for watermarking, focusing on embedding and concealing data within digital images to ensure intellectual property protection and secure communication. The study addresses the inherent challenges of watermarking, particularly the need for robustness and imperceptibility of the embedded watermarks. An advanced Transformed Least Significant Bit (LSB) embedding technique is presented as an effective solution for these goals. To meet the demands of real-time processing efficiency, we propose implementing watermarking algorithms in reconfigurable hardware. An architecture for the Transformed LSB watermarking algorithm is presented, specifically designed to optimize Field-Programmable Gate Array (FPGA) utilization while maintaining high throughput. This system supports a configurable secret message, allowing dynamic adjustment of the embedded data. Our approach facilitates the seamless integration of watermarking techniques into image processing applications, ensuring reliable protection of intellectual property and secure communication channels. Experimental results demonstrate the effectiveness of the proposed architecture in encrypting multiple images sequentially, validating its practical applicability and performance.**

***Index Terms*— Image Steganography, FPGA , Watermarking, Verilog, HD.**

I. INTRODUCTION

In the current age of digital technology, it is of utmost importance to guarantee the secrecy and accuracy of sensitive data while it is being transmitted. Image sharing is extensively utilized in several domains such as the internet, social networking platforms, and military operations. Preserving the photos from unwanted access has become extremely important. In the growing age of electronic commerce, it is necessary to address the issue of guaranteeing secrecy and data authentication in the current fast-paced and open communication setting. Data concealing methods provide good data authentication and make it difficult for hackers to recover the original image. Requirements such as authentication, integrity, secrecy and availability need to be considered for secure transfer of information via the usage of the stego pictures. The increasing number of digital communication channels has heightened the demand for strong encryption and steganographic methods. Steganography is the practice of hiding confidential information within harmless media, providing a hidden method of communication that complements conventional cryptographic techniques. Digital images have become popular for hiding information because they are widely used and have a complex structure.
The introduction of Field-Programmable Gate Arrays (FPGAs) has completely transformed the field of digital hardware design, providing unmatched flexibility and performance. FPGAs are integrated circuits that can be

Grenze ID: 01.GIJET.11.1.817

programmed to create custom digital logic circuits, making them well-suited for speeding up computationally demanding tasks. Image steganography, which involves extensive mathematical transformations and processes for embedding data, can greatly benefit from the use of FPGA-based acceleration.
In this paper, a security scheme with steganography based on LSB manipulation is shown. The details of the working methods, application and results are explained in the next sections.

II. Literature Review

At present, architectural designs are utilized in several image processing applications like pre-processing [1], segmentation [2], motion estimation [3], and more. Creating hardware for image steganography algorithms and delivering real-time results is a challenging task because of the computational complexity of the concealing and extraction processes. In the spatial domain schemes, secret data is directly hidden over the pixel values of the cover image. For LSB methods, the least significant bits of pixels in the cover image are discarded and replaced by secret data.
The advantages of spatial domain approaches include straightforward implementation and rapid operating performance. Nevertheless, these technologies often exhibit lower levels of reliability in terms of robustness and security [4]. It is revealed that the conventional LSB approach is vulnerable to RS attack [5]. To boost security, LSB matching (LSBM) is suggested and developed [6]. Unlike the ordinary LSB replacement, in the LSB matching, the pixel values are increased or lowered arbitrarily by one to match the messages to be buried. The LSBM minimizes the imbalance in the embedding distortion and hence demonstrates improved security in repelling steganographic assaults.
Edgar et al. [2] provides hardware architecture of a steganographic context approach appropriate for FPGA implementation. The context approach exploits noisy sections as well as quick gray level shifts in an image where the information is difficult to identify. The process to detect these regions in a picture is extremely regular and difficult from a computation point of view. Therefore, authors have chosen FPGA devices to accomplish design.
Rajgopalan et al. [7] developed the FPGA implementation of a steganographic method that randomizes capacity of data contained in every pixel, according to the comparable MSBs. Three distinct strategies have been presented based on indication selection among RGB channels. It has been established that increase in picture size increases the message embedding time and number of logic pieces. The throughput attained in this approach for a color image size of 256 × 256 pixels is 5.898 ms or 2,94,900 clock cycles of 50 MHz. Although, the experimental results shown for various color photos assures that this technique works well for medium to big size images, inadequate hardware implementation specifics indicates that reaching a high throughput is quite difficult
Yang and co-workers [8] suggested a strategy with adaptive LSB substitution via which they got a high embedding rate and good invisibility of the stego-image. In this method, they use the edge masking features to acquire the higher-order picture by using the higher-order bits of the original image and applying the adaptive LSB method thus a new steganographic image methodology is developed. Jose and co-workers [9] suggested a steganographic approach predicated upon a 3-bit data concealment technique that depends on LSB. In the suggested approach, 3-bits of message and picture is taken at a time and is inserted to LSB of the cover image. By utilizing this approach the sequence mapping may be avoided and it makes it difficult for the hackers to decipher the associated steganographic image. Sharma and Kumar [10] created a modified LSB approach which is predicated upon the idea that the pixels consist of the combination of RGB. In this way, the data concealing has been secured by changing RGB components. The initial byte of secret information is placed within the first-pixel red component. Then, the second byte of the secret information is inserted into the second-pixel green component.
The explored papers showcase significant progress in various image steganographic techniques, like LSB, MSB manipulation at the forefront. However, their computational demands pose challenges for deployment in an FPGA that is resource constrained. In this paper, a security scheme with steganography based on LSB manipulation is shown. The details of the working methods, application and results are explained in the next sections.

III. METHODOLOGY

The proposed methodology consists of two primary modules (i) Transpose Module and (ii) LSB manipulation module as shown in Fig. 1. This section provides a detailed description of the implementation process, focusing on the development of these modules and their integration to create a functional steganographic system.

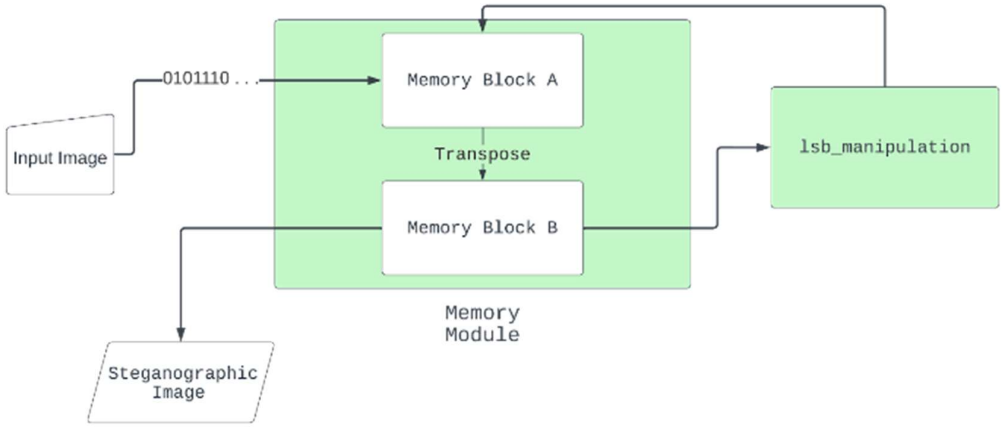

Figure 1. System Architecture of Image Steganography Module

The image is given as an input to the image transpose module. This module transposes the input image. It rearranges the rows and columns of the image. The output from the transpose module, which is the transposed version of the original image, is passed to the LSB(Least Significant Bit) manipulation block. This module performs the LSB manipulation to hide the secret data into the transposed image. The output from the LSB manipulation block module, which is the image with hidden secret data i.e steganographic image.

A. Image Transpose

The image transpose module plays a crucial role within the proposed image steganographic system, serving as a foundational component responsible for manipulating the pixel arrangement of the input image. Its significance lies in its ability to initiate the steganographic process by introducing alterations to the image structure before embedding confidential data. This module comprises two 3D memory blocks, denoted as A and B, each with dimensions of 8x100x100. Memory block A acts as the repository for the original image, while memory block B serves as the container for its transposed counterpart. It's worth noting that the color information within each pixel can be represented in grayscale, offering flexibility in data representation.

The process of transposing the image entails a fundamental operation of exchanging rows with columns, a transformation that is pivotal in reshaping the visual representation of the image. This operation not only modifies the spatial arrangement of pixel values but also contributes to the concealment process by introducing a structural modification that is imperceptible to the naked eye. Furthermore, the implementation of this transformation underscores the module's adaptability to diverse image formats and sizes, thereby ensuring its applicability across a wide range of scenarios.

The mechanism for storing the image data within the module is facilitated through the utilization of the img_in port, which accommodates 8-bit input data. This input mechanism is integral to the seamless integration of the image data into the steganographic workflow. Additionally, the adoption of the SPI (Serial Peripheral Interface) protocol for data transfer underscores the module's efficiency in facilitating rapid and lossless transmission of image data, thereby enhancing the overall performance and reliability of the system.

Upon the completion of the image storage process within memory block A, the activation of the img_transfer_done signal serves as a trigger for initiating the image transpose logic. This logic orchestrates the seamless transfer of the image data from memory block A to its transposed location within memory block B. The efficiency of this transfer process is underscored by its ability to execute within a single clock cycle, thereby minimizing latency and optimizing system throughput.

Subsequently, the transposed image data undergoes further processing within the LSB (Least Significant Bit) Manipulation module, where the LSBs of each pixel are modified to embed the confidential data. This module operates in tandem with memory block B, facilitating the serial retrieval of cover data through its cover_data port, while simultaneously providing stego data through its stego_data port. This synchronized operation ensures

the seamless integration of confidential data into the image data stream, laying the groundwork for generating stego image data.

As the stego image data traverses through the LSB Manipulation module, it is stored within memory block A, culminating in completing the data embedding process. Concurrently, memory block B serves as the conduit for the transposition of the stego image data, thereby facilitating the generation of the final output. This output, which encapsulates the concealed confidential data within the transposed image structure, is made accessible through the img_out port, thereby concluding the steganographic process.

B. LSB Manipulation

In the LSB Manipulation, the module contains a 1D secret_data memory, where secret data is stored via the secret_message port using SPI protocol. This memory awaits incoming image data through the cover_data port, transmitted by the transpose module. Upon receiving data, the LSB of each pixel is extracted, and the corresponding secret message bit is appended. The resulting data is then outputted through the stego_data port. This process iterates over each pixel, allowing for the sequential embedding of secret data into the image data stream.

IV. RESULTS AND DISCUSSIONS

The image, originally 8-bit and sized 100x100, was converted to a hexadecimal file using a Python script. This hexadecimal representation was then sent pixel by pixel via SPI protocol to the transpose module in the testbench environment. The transpose module processed the image, resulting in a modified output image stored in another hexadecimal file. This process showcases the core functionality of the steganographic system, from image conversion to transposition and storage. The output image in hexadecimal format is critical for further evaluation, focusing on the quality and fidelity of the transposed image and potential system optimizations. The successful execution of these steps demonstrates the system's capability to handle and process image data according to specified protocols. Future analysis will aim to assess the effectiveness of the transposition and identify areas for improvement, ensuring the system maintains high standards of image quality and fidelity.

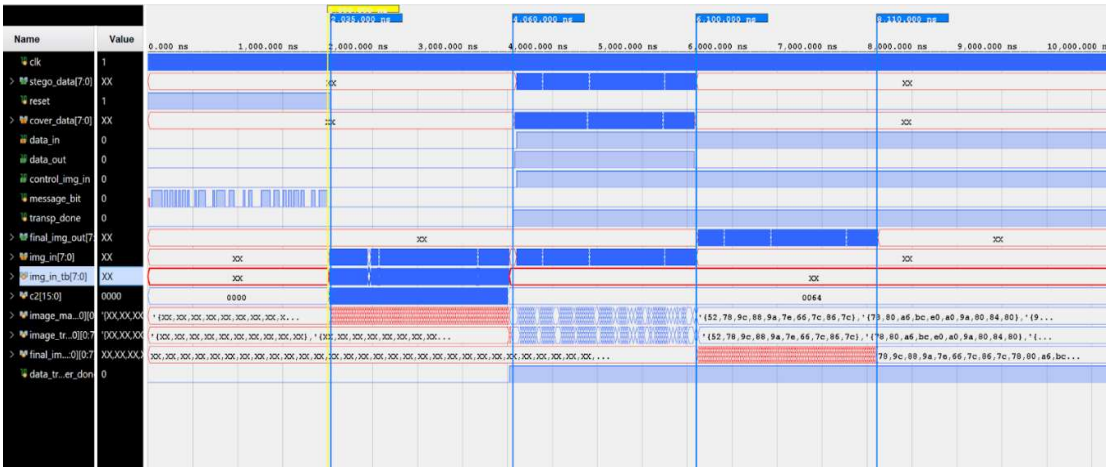

Figure 2. Simulation Waveform

The simulation waveform in Fig. 2 comprises four distinct phases. In the initial phase, the secret message is stored in the secret_message memory of the LSB Module.

The second phase involves transmitting image data, with 8 bits transferred per clock cycle, to the transpose module, which is stored in memory block A.

The image data is routed to the LSB Module during the third phase for processing. Simultaneously, the processed data is returned to the transpose module, indicating parallel processing.

In the final phase, the output image is received by the testbench and stored in a file shown in the Fig. 2 for further analysis.

Fig. 4 shows the original image, while Fig. 5 depicts the stego image. The impact on the original image is approximately 0.00002%, indicating an extremely low and imperceptible alteration. This demonstrates that the steganographic process preserves the image's visual quality while securely embedding hidden data.

Figure 3. Final output Image Hex File

Figure 4. Original Image

Figure 5. Stego Image

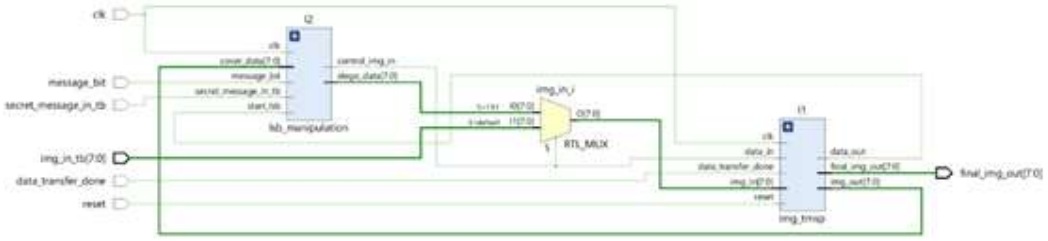

Figure 6. Elaborated Design

Fig. 6 illustrates the RTL schematic with 21 I/O pins: 13 for input and 8 for output. Among the 13 inputs, 4 are control signals (Clock, Reset, Enable, Select) that manage the system's operations. The 8 output pins handle processed data and status signals, indicating various system states.

1. Slice Logic

Site Type	Used	Fixed	Prohibited	Available	Util%
Slice LUTs	22915	0	0	64000	35.80
LUT as Logic	22915	0	0	64000	35.80
LUT as Memory	0	0	0	17600	0.00
Slice Registers	10180	0	0	128000	7.95
Register as Flip Flop	10180	0	0	128000	7.95
Register as Latch	0	0	0	128000	0.00
F7 Muxes	1333	0	0	32000	4.17
F8 Muxes	666	0	0	16000	4.16

* Warning! LUT value is adjusted to account for LUT combining.

Figure 7. Implementation report of Spartan 7 Board

The utilization report in Fig. 7 reveals a moderate usage of slice logic resources, with LUTs being moderately utilized at 35.80% and registers relatively lightly used at 7.95%. However, no memory, DSPs, or significant IO resources are utilized, suggesting potential for optimizing or reassessing design requirements.

V. Conclusion and Future Scope

The simulated steganographic system demonstrates robust functionality and efficient operation across all phases. It seamlessly handles image data and secret messages. The configurable nature allows for dynamic adjustment of embedded data, enhancing its adaptability for diverse applications. Furthermore, support for the SPI protocol enables seamless integration with external devices, expanding compatibility and communication capabilities.

The system's ability to continuously watermark multiple images offers practicality and efficiency in real-world scenarios, bolstering its utility in secure communication and intellectual property protection. Additionally, compatibility with any camera enhances versatility and applicability in diverse settings, catering to both consumer-grade and professional imaging devices. Looking ahead, further development and optimization efforts can refine performance metrics and explore advanced encryption techniques, enhancing system security and reliability. While SPI protocol support and camera compatibility are already functional, continued research and development can unlock the system's full potential and foster innovative applications across domains.

REFERENCES

[1] M. Genovese, P. Bifulco, D. De Caro, E. Napoli, N. Petra, M. Romano, M. Cesarelli, A.G.M. Strollo, Hardware implementation of a spatio-temporal average filter for real-time denoising of fluoroscopic images, Integration, Volume 49, 2015, Pages 114-124, ISSN 0167-9260.

[2] Gomez-Hernandez, Edgar & Feregrino, Claudia & Cumplido, Rene. (2008). FPGA Hardware Architecture of the Steganographic ConText Technique. 123 - 128. 10.1109/CONIELECOMP.2008.24.

[3] Hemalatha, R. J. et al. "Implementation of medical image segmentation using Virtex FPGA kit." 2015 International Conference on Signal Processing and Communication Engineering Systems (2015): 358-362.

[4] Rufeng Chu, Xinggang You, Xtangwei Kong and Xiaohui Ba, "A DCT-based image steganographic method resisting statistical attacks," 2004 IEEE International Conference on Acoustics, Speech, and Signal Processing, Montreal, QC, Canada, 2004, pp. V-953, doi: 10.1109/ICASSP.2004.1327270.

[5] J. Fridrich, M. Goljan and Rui Du, "Detecting LSB steganography in color, and gray-scale images," in IEEE MultiMedia, vol. 8, no. 4, pp. 22-28, Oct.-Dec. 2001, doi: 10.1109/93.959097

[6] J. Mielikainen, "LSB matching revisited," in IEEE Signal Processing Letters, vol. 13, no. 5, pp. 285-287, May 2006, doi: 10.1109/LSP.2006.870357.

[7] Rajagopala, Sundararaman et al. "MSB Based Embedding with Integrity: An Adaptive RGB Stego on FPGA Platform." Information Technology Journal 13 (2014): 1945-1952.

[8] C. -C. Chang and H. -W. Tseng, "Data Hiding in Images by Hybrid LSB Substitution," 2009 Third International Conference on Multimedia and Ubiquitous Engineering, Qingdao, China, 2009, pp. 360-363, doi: 10.1109/MUE.2009.68.

[9] Mekha Jose, "Hiding Image in Image Using LSB Insertion Method with Improved Security and Quality", International Journal of Science and Research (IJSR), Volume 3 Issue 9, September 2014, pp. 2281-2284,

[10] Ali, Abdelmgeid & Abd El-Hafeez, Tarek & Seddik, Al Hussien & M., Shaimaa. (2016). New Image Steganography Method using Zero Order Hold Zooming. International Journal of Computer Applications. 133. 27-31. 10.5120/ijca2016908016.

[11] H.-M. Sun, M.-E. Wu, W.-C. Ting, M. Hinek, "Dual RSA and itssecurity analysis. Information Theory" IEEE Transactions, 2007, vol.53, pp. 2922–2933.

[12] K. B. Padeppagol, M. H. Sandhya Rani, "Design and implementation of lifting based wavelet and adaptive LSB steganography to secret data sharing through image on FPGA" International Journal of Engineering and Management Research (IJEMR), 2018, vol. 8, pp. 112–119

[13] C.-C. Cang, H.-W. Tseng, "Data Hiding in Images by Hybrid LSB Substitution" 2009 Third International Conference on Multimedia and Ubiquitous Engineering, IEEE, pp. 360–363.

[14] M. Jose, "Hiding Image in Image Using LSB Insertion Method with Improved Security and Quality", IJSR, 2014, vol. 3, pp. 2281–2284.

[15] V. Sharma, S. Kumar, "A new approach to hide text in images using steganography", International Journal of Advanced Research in Computer Science and Software Engineering (IJARCSSE), 2013, vol. 3, pp. 701–708.

[16] Z.-A. Jamal, Ahmadabadi, M. S. Ebrahim, A. Latif, "An Adaptive Secret Image Sharing with a New Bitwise Steganographic Property" Information Sciences, 2016, vol. 369, pp. 467–480.

[17] X. Wu, C.-N. Yang,. (). Invertible secret image sharing with steganography and authentication for AMBTC compressed images. Signal Processing: Image Communication. 2019, vol. 78. pp. 437–447.

[18] "A block based data hiding method in images using pixel value differencing and LSB substitution method", Tasnuva Mahjabin, Syed

[19] Monowar Hossain, Md.Shariful Haque, 15th International Conference on Computers and Information Technology,pp.168-172

[20] "An adaptive secret image sharing with a new bitwise steganographic property" Jamal Zarepour-Ahmadabadi ,MohammadEbrahim Shiri Ahmadabadi, AliMohammad Latif- ELSEVIER volume:369.