Computer Science • 23(4) 2022 https://doi.org/10.7494/csci.2022.23.4.4491

Sunil B. Hebbale
Dr. V.S. Giridhar Akula
Dr. Parashuram Baraki

FPGA-BASED SECURE AND NOISELESS IMAGE TRANSMISSION USING LEA AND OPTIMIZED BILATERAL FILTER

Abstract *In today's world, the transmission of secured and noiseless images is a difficult task. Therefore, effective strategies are important for securing data or secret images from attackers. Besides, denoising approaches are important for obtaining noise-free images. For this, an effective crypto-steganography method that is based on a lightweight encryption algorithm (LEA) and the modified least significant bit (MLSB) method for secured transmission is proposed. Moreover, a bilateral filter-based whale optimization algorithm (WOA) is used for image denoising. Before the image transmission, a secret image is encrypted by the LEA algorithm and embedded into the cover image using discrete wavelet transform (DWT) and MLSB techniques. After the image transmission, an extraction process is performed in order to recover the secret image. Finally, a bilateral WOA filter is used to remove the noise from the secret image. The Verilog code for the proposed model is designed and simulated in Xilinx software. Finally, the simulation results show that the proposed filtering technique results in performance that is superior to conventional bilateral and Gaussian filters in terms of the peak signal-to-noise ratio (PSNR) and structural similarity index measure (SSIM).*

Keywords lightweight encryption algorithm, bilateral filter, whale optimization algorithm, discrete wavelet transform

Citation Computer Science 23(4) 2022: 451–466

1. Introduction

Digital images are used in various fields like medical diagnostics and entertainment as well as in other scientific and engineering applications. Astronomical images, satellite images, CT scans, and MRI images are some examples of digital images that have become integral parts of our lives [21], [7], [9] Communication technologies such as internet networks and mobile phones have advanced rapidly in the modern world, and the area of information processing has expanded to include visual communications such as videos and photographs. During the communication process, the data is transmitted from one end to another. At that time there will be a high risk of data-stealing or may get altered [12], [1], [20] To prevent the data from this unauthorized access, several security techniques are provided. Moreover, noise occurrence of these images is another important concept [17], [22]. Generally, the noises affect the images during the transmission or acquisition process. The quality of the images may be affected when the images are transmitted through an error-prone channel. Therefore, providing security and removing noise from these images have essential concerns [14].

In the last two decades, various steganography and cryptography techniques are created to deal with data security named Rivest Shamir Adleman Advanced Encryption Standard etc, [13], [16], [10]. However, these algorithms have very high processing time. Moreover, various filtering approaches such as Gaussian filter [23], median filter are provided by various researches for image denoising which are based on different noise models like speckle, salt and pepper,etc. [25].

However, an efficient denoising algorithm should have the ability to eliminate noise while retaining the image's most informative features, such as corners, sharp structures and edges. And also, maximum researches concentrated only on security or image demonising [19], [11]. For this reason, the present paper targeted to concentrate on both image security and denoising. For security, both cryptography (LEA) and steganography (DWT, MLSB) approaches are presented in this paper. LEA is a powerful cryptography algorithm that makes data more secure and less susceptible to attacks. In addition, it is faster and takes less memory space on several platforms than other encryption algorithms [24], [27], [3]. Moreover, for image steganography, DWT and MLSB methods are used. For image denoising, a whale optimization-based bilateral filter is used. It is a well-known approach. It is commonly used in a variety of image processing applications. It removes the noise while preserving the image's important information [2]. Moreover, the whale optimization algorithm has the advantages of high classification accuracy, easy operation, and few parameters. The proposed technique is designed in software-based approaches and it can be implemented in hardware. By means of certain metrics such as decreased execution time, few area and reduced power consumption, the hardware-based methods have an advantage over software-based methods. Hardware-based techniques utilized in a wide range of applications and satisfy the majority of real-time specifications. It provides high processing speed, capable of connecting with other devices, and portability. For hardware-based methods, different platforms are available such as Application Specific

Integrated Circuit (ASIC), Field Programmable Gate Array (FPGA), Digital Signal Processing (DSP), and General Purpose Processor (GPP). Among these FPGA is a widely used technique which is less expensive, reconfigurable and flexible than ASICs. The proposed approach is designed using Verilog code. Hence it can be easily implemented on FPGA.

The main contributions of the proposed approach are as follows:

1) To design an effective encryption system based on a lightweight encryption algorithm and a modified LSB method for secured image transmission; also, the proposed model can be implemented in FPGA for real-world applications.

2) To design an effective filtering technique based on a bilateral filter with a whale optimization algorithm for obtaining noiseless images. To evaluate the performance of the proposed model, the following parameters were taken: PSNR, and SSIM. The rest of the paper is arranged as follows: we analyze recent studies about noiseless and secure image transmission in Section 2; a detailed problem statement is presented in Section 3; the proposed methodology is presented in Section 4; Section 5 provides the implementation details of FPGA; Section 6 provides the obtained results and its explanation; and Section 7 describes our conclusions.

2. Literature review

Some recent works that are related to secure and noiseless image transmission are listed in this section. An image-encryption algorithm was suggested by Yang and Chien [26] that was based on improved AES and a four-dimensional chaotic system. In FPGA, parallel and pipeline computing features were used to optimize the encryption algorithm. Primarily, the keys were generated by the chaotic system. Then, SubBytes and shift rows of improved AES were modified with cubic S-Box and Spin-Sort, and the round of encryption was decreased. For the hardware implementation, FPGA was used by the authors; they used information entropy, differential attack, correlation, and histogram analyses for their performance estimation.

An image-encryption model that was developed by Hafsa et al. [5]was based on modified AES. A pseudorandom number generator (PRNG) was used for the key-generation process. In this process, two different chaotic systems were included with PRNG. Moreover, four different S-boxes were used for the SubBytes operation. The random permutation approach was used to replace the transformation of the mix-columns and shift rows; this increased the complexity. For their performance evaluation, randomness and histogram analyses were used. Two image-steganography algorithms were suggested by Ismail et al. [6] that were based on generalized chaotic maps. In the first algorithm, only one chaotic map was utilized for one-dimensional memory indexing; for the two-dimensional memory indexing, two chaotic maps were utilized in the second algorithm. For the selecting a cover image pixel's position, pseudorandom numbers were implemented. Moreover, generalized logistic maps were used for the designs of the different chaotic behaviors; this increased the security level of the suggested approach. For their performance validation, four different image

sets were used. Image fidelity (IF), normalized cross-correlation (NCC), structural similarity index (SSIM), mean squared error (MSE), and peak signal-to-noise ratio (PSNR) were utilized for their performance comparisons.

An approximated fractional integrator-based denoising algorithm (AFI) was developed by Kumar and Jha [8]. Two types of images (grayscale and binary images) were considered for their experiment. For noisy binary images, the filter/mask (of different orders of q) was explicitly implemented. Furthermore, the fractional-order q approach was implemented for grayscale images. For their performance assessment, cross-correlation, the structural similarity index, and the peak-signal-to-noise ratio were used as the performance metrics, and some existing techniques were implemented for the comparison.

For image denoising, modified recursive box filter-based fast BF (MRBF) was implemented by Bhargava and Gangadharan [4]. In MRBF, a modified carry-select adder was utilized that was based on quantum-dot cellular automata technology. The additive white Gaussian noise model was selected for this approach. Standard grayscale images were used for their performance evaluation, and SSIM and PSNR were utilized as the performance metrics.

Exploiting modification direction-based image steganography algorithms (EMDs) were developed by Shet et al. [18]. They used FPGA for their hardware implementation. Full EMD (FEMD), basic EMD (BEMD), and modulus operation-based EMD (MEMD) were the three variants of the EMD steganography methods that were implemented by the authors. To improve the operational speed, parallel and pipelining processing were used in each of the modules. For their performance estimation, the MSE, PSNR, and MSSIM performance metrics were used. To suppress any Gaussian and impulsive noise, Prajapati and Darji [15] developed a step-size scalar-based adaptive filter with a modified robust mixed norm (MRMN). To achieve a better convergence rate, an LMS adaptive algorithm was used. An instant-switching mechanism with a constant switching threshold of α was utilized for error reduction. The MSE minimization, convergence rate, and SSE minimization performance metrics were used. Different adaptive algorithms were used for the performance comparison.

3. Problem statement

Security and noise are two key issues that occur during the image-transmission process. During the image- or data-transmission process over a network, security is critical. In order to maintain the security of the data, it must be encrypted; in this way, an intruder cannot hack the data nor be capable of understanding it. Sometimes, the quality of a transmitted image is decreased by the noisy channel and encryption process; therefore, image noise reduction is essential for retrieving high-quality images from noisy (degraded) observations. In computer vision and image processing, this is one of the most fundamental and well-known issues. During the image-denoising process, significant difficulties such as preserving textures, generating new artifacts, protecting edges against blurring, and maintaining smooth flat areas should be top

priorities. Therefore, an effective encryption and denoising approach is presented in this paper for secure and noiseless image transmission.

4. Proposed methodology

This section presents our secure noiseless image transmission model; its proposed architecture is shown in Fig 1.

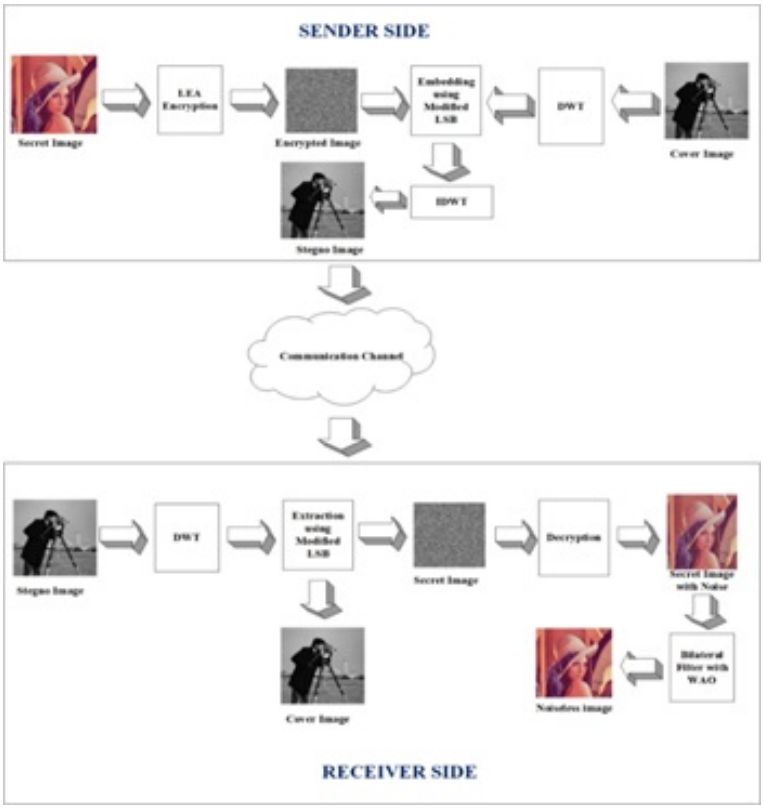

Figure 1. Proposed Architecture

Initially, two images are selected for the cover and secret images (128 x 128 and 64 x 64 pixels, respectively). Then, MATLAB is used to convert these images into a binary format. During this process, binary text files are generated for the cover and secret images. The binary files are received as inputs by the sender module through test cases. In the sender module, the LEA algorithm is used to encrypt the secret image's data. Additionally, DWT is applied to the cover image's data in order to produce high- and low-frequency elements. Then, the embedded module uses the modified LSB technique to embed the encrypted data with high pass coefficients and

then applies the inverse scaling operation of DWT to generate a stegano image (128 x 128); this is the embedded version of the hidden secret image with a cover image.

To recover the hidden image, the receiver module uses the reverse procedure of the embedding process. For this, DWT is initially applied to the stegano image in order to generate the low- and high-frequency components. After this, these frequency components are extracted by the modified LSB approach in order to recover the secret image bits (which are in an encrypted form). To perform the decryption, an LEA algorithm with the same key is used. Then, the filtering process is carried out to remove the noise from the secret data. For this, a bilateral filter with the whale optimization algorithm is implemented. After this, the binary text file is converted into an image by using MATLAB. A detailed explanation of the proposed model is given below.

4.1. Lightweight encryption algorithm on secret image

The LEA encryption algorithm is a symmetric key cipher with a 128-bit block size. Initially, the hidden image data is divided into several blocks. We assume that $H_i = (H_i[0], H_i[1], H_i[2], H_i[3])$are the input blocks (secret image data) of the i^th iteration, and each block has a 32-bit size. $K_0, K_1, K_2, K_3, K_4, K_5$ represent the round keys (Ki); each round key has 32 bits. To generate the round keys, various constant values are utilized (A = [0] through [7]). The constant values are as follows:

A [0] = 0x3efe9db;
A [1] = 0x44626b02;
A [2] = 0x79e27c8a;
A [3] = 0x78df30ec;
A [4] = 0x715ea49e;
A [5] = 0xc785da0a;
A [6] = 0xe04ef22a;
A [7] = 0xe5c40957.

These constant values are generated from 766995 (which is the hexadecimal expression). Here, "76" denotes the ASCII code of the letter (L), "69" denotes (E), and "95" denotes (A). Using these constant values, the key-generation process of the LEA algorithm is conducted, which is described as follows:

$K[0] \leftarrow RL_1(K[0] + RL_i(A[imod4]);$
$K[1] \leftarrow RL_3(K[1] + RL_{i+1}(A[imod4]);$
$K[2] \leftarrow RL_6(K[2] + RL_{i+2}(A[imod4]);$
$K[3] \leftarrow RL_11(K[3] + RL_{i+3}(A[imod4]);$
$K_i \leftarrow (K[0], K[1], K[2], K[1], K[3], K[1]).$

Here, RL represents the left rotation, and (+) denotes the modular addition. For the 128-bit key size, the key-scheduling process used ($0 \leq i \leq 24$) rounds to generate round key Ki. This architecture repeats the process for a specified number of rounds, and the obtained values are stored as key values. After the round key generation, the

encryption process is executed. In this stage, cipher text C= (C[0], C[1], C[2], C[3]) is obtained from the plain text Hi (input image blocks) by the use of the round keys. The encryption process is carried out for the i^th round ($0 \leq$ i $\leq$ r- 1, r=24), which is described as follows:

$H_{i+1}[0] \leftarrow RL9\left((H_i[0] \oplus RK[0]) + (H_i[1] \oplus RK_i[1])\right)$;
$H_{i+1}[1] \leftarrow RR5\left((H_i[1] \oplus RK[2]) + (H_i[2] \oplus RK_i[3])\right)$;
$H_{i+1}[2] \leftarrow RR3\left((H_i[2] \oplus RK[4]) + (H_i[3] \oplus RK_i[5])\right)$;
$C[0] \leftarrow H_r[0], C[1] \leftarrow H_r[1], C[2] \leftarrow H_r[2], C[3] \leftarrow H_r[3]$.

Here, $\oplus$ is the XOR function, and + is a modular addition function. The above process is carried out for 24 rounds. The cipher text is obtained at the end of the round.

4.2. DWT process on cover image

In this section, DWT is used to decompose cover image data into sub-bands. Generally, a cover image is in the spatial domain; however, an unauthorized user can easily find a hidden image in the spatial domain. Therefore, DWT is implemented to represent the cover image with high- and low-frequency coefficients in the frequency domain. Furthermore, this offers the required resolution and pixel-by-pixel extraction of the high- and low-frequency bands. Subsampling and DWT are used to describe a cover image in the frequency domain; this gives a higher level of robustness. The obtained coefficients of the bands are as follows:

$$[CI_1 CI_2 CI_3 CI_4] = DWT(CI). \tag{1}$$

Here, CI2, CI3, and CI4 are the high-frequency bands, CI1 is the low-frequency band, and the cover image is denoted by CI. The high-frequency bands carry unimportant information such as the edges and texture of an image, and the low-frequency sub-band contains all of the important information of the image. All of the sub-bands have the same dimensions (64 x 64). Low- and high-frequency sub bands such as CI1 and CI4 are chosen for the next process. To obtain more high-frequency bands, DWT is again applied to the CI4 and CI1 bands. The extracted coefficients are shown as follows:

$$\left[\begin{array}{cccc} CI_1^{LL} & CI_1^{LH} & CI_1^{HL} & CI_1^{HH} \end{array}\right] = DWT\left(CI_1\right); \tag{2}$$

$$\left[CI_4^{LL} CI_4^{LH} CI_4^{HL} CI_4^{HH}\right] = DWT\left(CI_4\right). \tag{3}$$

Here, CI_4^{LL} and CI_1^{LL} are the low-frequency bands of [CI_4 and CI_1], respectively. Moreover, the high-frequency bands that are obtained from CI_4 and CI_1 are CII_4^{LH}, CI_4^{HL}, CI_4^{HH}, CI_1^{LH}, C_1^{HL}, and C_1^{HH}. After this, the high-frequency bands are selected in order to perform the image-steganography process.

4.3. Embedding module

In the embedding module, the MLSB approach is implemented in order to include the encrypted data of a secret image into a cover image's DWT coefficients. Generally,

the least-significant bit of the secret image is used to replace the cover image's least-significant bit with the LSB technique. In the proposed approach, the last two bits of each pixel in the DWT coefficients are replaced with the first two bits of the secret image bits. For example, the cover images binary format is 00100111 11101001 11001000 00100111 11001000 11101001 11001000 00100111, and the binary format of the secret image is 10010000 01100101. This is serially copied from the left-hand side of the secret image bit (MSB bit) and replaced with the cover image's LSB bit. Therefore, the output binary bit pattern is 00100110 11101001 11001000 00100100 11001001 11101010 11001001 00100101. To embed the secret image, replacing the last two bits of each pixel makes the changes in the value of its color (however, it cannot be identified by the human eye). Moreover, these variations are quite minimal. Therefore, the modifications to the cover image are minor and can only be detected and evaluated by comparing the histograms of the original image and the stegno-image. After this, the inverse scaling operation of DWT is implemented to attain the stegno picture when all the bits of the image are replaced. Then, the stegno image is transferred to the receiver end.

4.4. Extraction module

To obtain the secret and cover images separately, the DWT process is again applied to the stegno image to obtain the low- and high-frequency components. This procedure is identical to that of the stegano module. After the extraction process of the low- and high-frequency elements, the 2-bit LSB data is extracted from the image and integrated to obtain the encrypted secret image. Afterward, the decryption approach is implemented in order to obtain the secret image (which is the reverse procedure of the encryption). By using the round function and round keys, this converts the cipher image into a plain image. The decryption round keys are produced in the same way as the encryption round keys. Moreover, the rotation operation is implemented in reverse order, and the modular subtraction is executed.

4.5. Noise removal using bilateral filter

Basically, noises occur in an image during the transmission and acquisition processes. In the proposed method, a hidden image is denoised by using a bilateral filter, and its parameters are optimized by using a nature-inspired optimization technique that is known as the whale optimization algorithm. The performance of the bilateral filter depends on the σd and σr parameters (which stand for the spatial parameter and range parameter, respectively). Manually tuning these parameters for each picture is difficult and time-consuming. to resolve this, optimal values are needed for the image-filtering parameters. To accomplish this, nature-inspired optimization algorithms are utilized in order to find the best parameters. The bilateral filter's key logic is that two pixels are known to be near each other – not only when they appear in neighboring places, but also when their photometric ranges are identical. To conserve the edges of the image, the bilateral filter makes use of intensity variations. In a local

neighborhood, this filter computes the sum of the pixels' weights. To replace the value of a pixel, a weighted average is used in each neighboring pixel. The efficiency of the proposed filter for pixel x can be expressed by using Eq. (4):

$$I_x(x=p) = \frac{1}{W_p} \sum_{q \in S} g_s(\|p-q\|) f_{r^r} \left(I_P - I_q\right) I_q. \tag{4}$$

Here, s denotes the spatial neighborhood of Ix(x) and the pixel coordinates are denoted by q and p. The intensity Gaussian is denoted by fr and the spatial Gaussian is denoted by gs, which decreases the influence of distant pixels. Normalization factor Wp is calculated by using Eq. (5):

$$W_P = \sum_{q \in S} g_S(\|p-q\|) f_r \left(I_P - I_q\right), \tag{5}$$

where spatial kernel gs is used for smoothing the coordinate variations, range kernel fr is employed for smoothing the intensity variations, and q and p are the pixels' coordinates. The spatial neighborhood of Ix(x) is represented by S. Consider the coordinates of pixel (i,j) and its neighboring pixel coordinates (k,l) (which have been denoised by the suggested filter). To denoise pixel (i,j), Eq. (6) is used to decide the weight that is to be assigned to pixel (i,j):

$$w(i,j,k,l) = \exp\left(-\frac{(i-k)^2 + (j-l)^2}{2\sigma_d^2} - \frac{\|I(i,j) - I(k,l)\|^2}{2\sigma_r^2}\right). \tag{6}$$

Here, smoothing parameters σ_d and σ_r represent the spatial and intensity domains' behavior of bilateral filter, respectively, and the pixel intensities are denoted by I(k,l) and I(i,j). If the values of σ_d and σ_r are too high, then the significant elements may become over-smoothed. Moreover, the values of σ_r fluctuate with the noise variance, but the values of σ_d do not fluctuate with the variance of the noise. In order to obtain high-quality images, setting the values of these parameters is very important; therefore, a whale optimization algorithm is used to optimize the control parameters of the bilateral filter.

4.6. Whale optimization algorithm-based parameter optimization of bilateral filter

To optimize the control parameters of bilateral filters σ_r and σ_d, whale optimization is used. Initially, the population of whale and the control parameter's search spaces $\sigma_{\mathrm{r}} = [1, 16]\sigma_{\mathrm{d}} = [0.1, 10], \mathrm{X_i} = \{\sigma_{\mathrm{ri}}, and\sigma_{\mathrm{di}}\}$ are initialized. Here, i denotes the number of whales, and the solution of the optimization problem is the prey. Primarily, the initial positions of the whale (search agent) are assigned, and the fitness value of each whale is computed. In each iteration, the position of the whale is updated by using Eq. (7):

$$X_i(t+1) = X_i(t) - \Psi^* G. \tag{7}$$

In Eq. (7), the value of G is obtained by using Eq. (8):

$$\mathrm{G} = |\zeta \cdot X_i * (\mathrm{t}) - X_i(\mathrm{t})| \,. \tag{8}$$

Here, the coefficient vectors are denoted by ζ and ψ, the position vector is denoted by X, and X$*$ is the best location vector that has been obtained so far. Moreover, the value of coefficient vectors ζ and ψ are computed as $\zeta = 2 \cdot r$ and $\psi = 2\mathrm{u} \cdot \mathrm{r}$, respectively. Here, u is linearly reduced from 2 to 0, and the value of random vector r is within a range of [0, 1].

At the end of each iteration, each search agent's fitness value is calculated. The computed fitness rate is compared to the current fitness rate, and the best solution is updated. Finally, the best fitness value is evaluated in order to find the optimum solution.

5. FPGA implementation

The key components of the proposed approach are the sender and receiver modules. Encryption, DWT splitting, a modified LSB-based embedded module, and inverse DWT section are included in the sender part. The pixel data of the cover image (128 x 128) and secret image (64 x 64) is stored in 16,384 and 4096 memory locations (RAM), respectively. Each location has 32-bit pixels. After initializing the RAM with the input data, the control unit begins the process by transmitting the addresses to the RAM (the secret image and cover image data). After obtaining the address, the RAM sends secret image bytes to the encryption block and cover image data to the DWT block. The encryption is performed by using the LEA algorithm and a 32-bit key. During the process of the 128-bit key creation, one cycle is needed to keep the initial plain text. After this, 24 cycles are used to encrypt the data and store it in the RAM (a total of 24 rounds). Consequently, the filter coefficients are used in the DWT module to perform the scaling and inverse scaling processes. At the end of this process, the low- and high-frequency components are obtained. For further processing, the high-frequency components are selected. Next, the encrypted data and high-pass components are sent to the embedded module to perform the embedding. During this process, the high-pass component's LSB 2-bit data is selected one-by-one and replaced with the 2-bit MSB of the encrypted data. This procedure is continued until the final high-pass element is replaced. To acquire a stegno image, the inverse-scaling process is performed by the inverse DWT. Afterward, the resultant stegno image is transmitted to the RAM by the embedding module, and the address of the stegno image data is specified by the control unit.

In the receiver module, the control unit begins the process by transferring the address of the stegno image bytes to the RAM and sending it into the DWT block. In this module, the low- and high-frequency components are extracted, and the high-pass components are serially shifted one-by-one to extract the 2-bit LSB data; it is then stored in the RAM. Next, all of the obtained bits are concatenated to each other

and sent into the decryption module. Then, the decryption process is performed, and the secret image is obtained. Consequently, the filtering process is performed by the bilateral filter with the whale optimization algorithm in order to attain the noiseless image.

6. Experimental results

Initially, the conversion of the cover image and secret image data into a text file is performed by using MATLAB (Ver. 2015a). Afterward, the converted text files are given as input to the Verilog code. The complete architecture of the proposed work is coded into the hardware description language (Verilog); hence, it can apply to FPGA. For this, Verilog-HDL is utilized in the Xilinx 14.7 environment, and the ISIM tool is used to perform the simulation process. The proposed approach includes the sender and receiver modules. The simulation results of the sender and receiver modules are displayed in Figures 2 and 3, respectively.

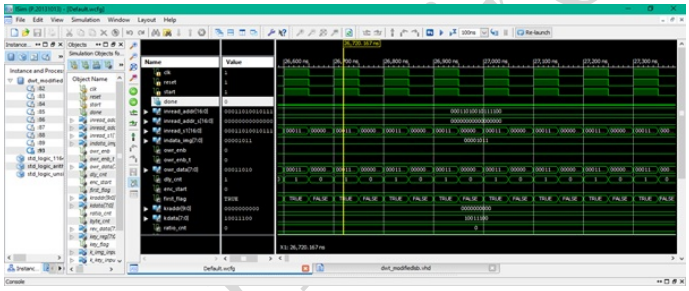

Figure 2. Waveform for top module of crypto-steganography (sender side)

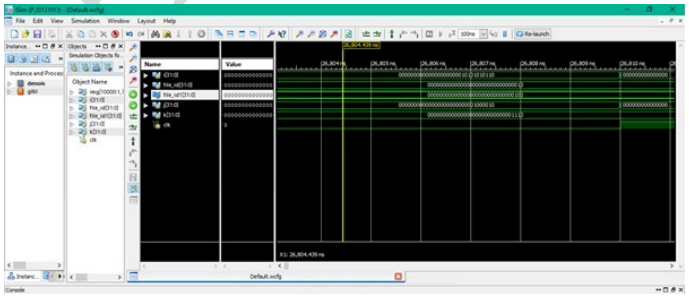

Figure 3. Output waveform of receiver-side image recreation and denoising (receiver side)

The input cover and secret image arrive after 40 ns for the sender module (as shown in the simulation waveforms of the proposed process). The system clock is denoted as 'clock', and the reset denotes the global reset signal that is utilized to

reset the system under power-on circumstances. Figure 2 shows the output of the sender module (stego image), which arrives after 26,720 ns. From the figure, it can be observed that the device embeds the hidden image into the cover image after 26,720 ns. Also, Figure 3 displays the retrieval of the secret image and noise removal processes. Moreover, the simulation waveform shows that the device extracts the hidden image from the cover image and removes the noise after 26,804 ns.

6.1. Evaluation of security

To analyze the performance of our image crypto-steganography approach, we used PSNR as the performance metric. For a comparison of our crypto-steganography approach, we utilized existing state-of-art approaches (AES-LSB and RSA-LSB). The performance comparison of the proposed approach (crypto-steganography) is presented in the following figure.

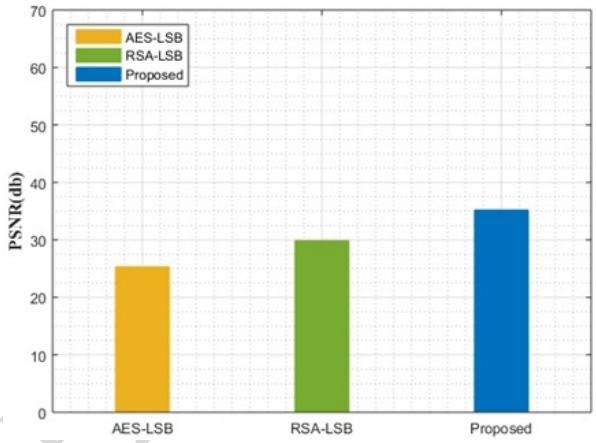

Figure 4. Performance comparison of PSNR value for crypto-steganography approach

From Figure 4, it can be observed that the proposed security approach is superior to the other approaches, as the PSNR value of the proposed approach is 35.12 (which is higher than with AES-LSB and RSA-LSB).

6.2. Evaluation of denoising

To measure the performance of the proposed denoising approach, two performance metrics were used: the peak signal-to-noise ratio (PSNR), and the structural similarity index (SSIM). Here, the image is corrupted by salt-and-pepper noise. For the performance comparison, existing state-of-the-art techniques like the Gaussian and median filter were chosen. Performance comparisons of the PSNR and SSIM values are shown in Figures 5 and 6, respectively.

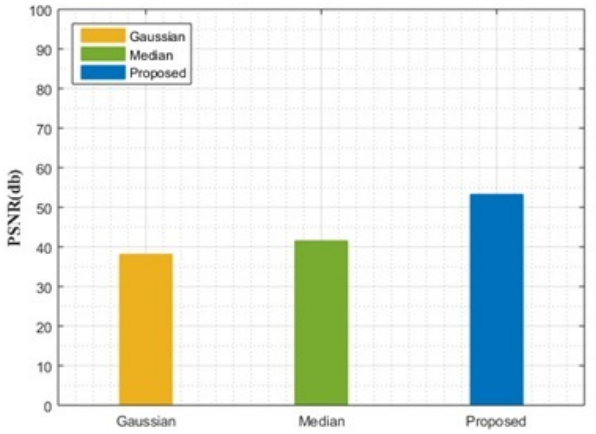

Figure 5. Comparison of PSNR values with existing and proposed approaches

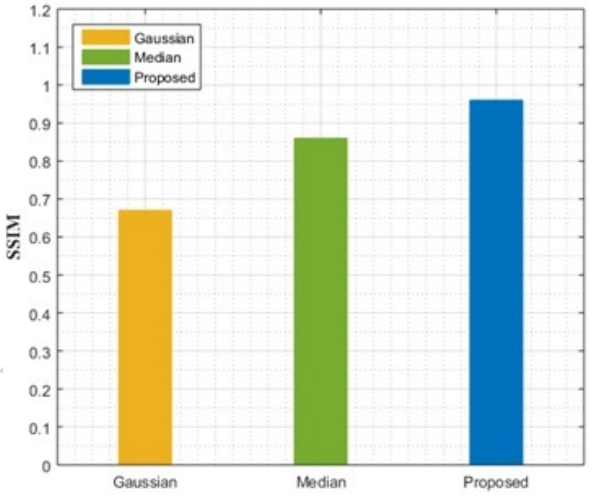

Figure 6. Comparison of SSIM values with existing and proposed approaches

The proposed FPGA architecture's denoised results are found to be much superior to the denoised results of the existing techniques. Moreover, the proposed architecture achieved an SSIM value of 0.96, and PSNR was 53.18 dB. This is significantly better than the current methods.

7. Conclusion

In this research work, an effective crypto-steganography approach with noiseless image transmission has been proposed. A cover image is used to embed a secret image

in the stegno module with superior security (LEA). In addition, the stegno image is successfully extracted, and the secret image is obtained in the recovery module. Moreover, the bilateral filter with WOA is effectively performed, and the noise from the image is removed without decreasing the quality of the image. A performance investigation of the crypto-steganography and denoising techniques has been estimated by the SSIM and PSNR performance metrics. When compared to other approaches, the performance of the proposed approach is superior; it provides high-level security and a noiseless image. In the future, the proposed work can be extended to implement audio- and video-based image steganography.

Acknowledgements

We declare that this manuscript is original, has not been published before, and is not currently being considered for publication elsewhere.

References

[1] Abdullah H.A., Abdullah H.N.: FPGA implementation of color image encryption using a new chaotic map, *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 13(1), pp. 129–137, 2019.

[2] ALabaichi A., Al-Dabbas M.A.A.K., Salih A.: Image steganography using least significant bit and secret map techniques., *International journal of electrical & computer engineering (2088-8708)*, vol. 10(1), 2020.

[3] Ansari A.S., Mohammadi M.S., Parvez M.T.: A multiple-format steganography algorithm for color images, *IEEE Access*, vol. 8, pp. 83926–83939, 2020.

[4] Bhargava G.U., Gangadharan S.V.: FPGA implementation of modified recursive box filter-based fast bilateral filter for image denoising, *Circuits, Systems, and Signal Processing*, vol. 40(3), pp. 1438–1457, 2021.

[5] Hafsa A., Gafsi M., Malek J., Machhout M.: FPGA implementation of improved security approach for medical image encryption and decryption, *Scientific Programming*, vol. 2021, 2021.

[6] Ismail S.M., Ghidan A.M., Zaki P.W.: Novel chaotic random memory indexing steganography on FPGA, *AEU-International Journal of Electronics and Communications*, vol. 125, p. 153367, 2020.

[7] Jang S.J., Hwang Y.: Noise-aware and light-weight VLSI design of bilateral filter for robust and fast image denoising in mobile systems, *Sensors*, vol. 20(17), p. 4722, 2020.

[8] Kumar S., Jha R.K.: An FPGA-based design for a real-time image denoising using approximated fractional integrator, *Multidimensional systems and signal processing*, vol. 31(4), pp. 1317–1339, 2020.

[9] Lien C.Y., Tang C.H., Chen P.Y., Kuo Y.T., Deng Y.L.: A low-cost VLSI architecture of the bilateral filter for real-time image denoising, *IEEE Access*, vol. 8, pp. 64278–64283, 2020.

[10] Madhusudhan K., Sakthivel P.: A secure medical image transmission algorithm based on binary bits and Arnold map, *Journal of Ambient Intelligence and Humanized Computing*, vol. 12(5), pp. 5413–5420, 2021.

[11] Malladi S.R.S., Ram S., Rodríguez J.J.: Image denoising using superpixel-based PCA, *IEEE Transactions on Multimedia*, vol. 23, pp. 2297–2309, 2020.

[12] Manoj Kumar T., Karthigaikumar P.: FPGA implementation of an optimized key expansion module of AES algorithm for secure transmission of personal ECG signals, *Design Automation for Embedded Systems*, vol. 22(1), pp. 13–24, 2018.

[13] Marwan M., Dos Santos V., Abidin M.Z., Xiong A.: Coexisting Attractor in a Gyrostat Chaotic System via Basin of Attraction and Synchronization of Two Nonidentical Mechanical Systems, *Mathematics*, vol. 10(11), p. 1914, 2022.

[14] Phadikar A., Maity G.K., Chiu T.L., Mandal H.: FPGA implementation of lifting-based data hiding scheme for efficient quality access control of images, *Circuits, Systems, and Signal Processing*, vol. 38(2), pp. 847–873, 2019.

[15] Prajapati P.H., Darji A.D.: FPGA implementation of MRMN with step-size scaler adaptive filter for impulsive noise reduction, *Circuits, Systems, and Signal Processing*, vol. 39(7), pp. 3682–3710, 2020.

[16] Setyono A., Muljono M., *et al.*: Dual encryption techniques for secure image transmission, *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)*, vol. 10(3-2), pp. 41–46, 2018.

[17] Sheela C.J.J., Suganthi G.: An efficient denoising of impulse noise from MRI using adaptive switching modified decision based unsymmetric trimmed median filter, *Biomedical Signal Processing and Control*, vol. 55, p. 101657, 2020.

[18] Shet K.S., Aswath A., Hanumantharaju M., Gao X.Z.: Novel high-speed reconfigurable FPGA architectures for EMD-based image steganography, *Multimedia Tools and Applications*, vol. 78(13), pp. 18309–18338, 2019.

[19] Shukla A.K., Pandey R.K., Yadav S., Pachori R.B.: Generalized fractional filter-based algorithm for image denoising, *Circuits, Systems, and Signal Processing*, vol. 39(1), pp. 363–390, 2020.

[20] Soleimani Abhari P., Razaghian F.: A novel median based image impulse noise suppression system using spiking neurons on FPGA, *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*, vol. 8(6), pp. 631–640, 2020.

[21] Taghinia Jelodari P., Parsa Kordasiabi M., Sheikhaei S., Forouzandeh B.: FPGA implementation of an adaptive window size image impulse noise suppression system, *Journal of Real-Time Image Processing*, vol. 16(6), pp. 2015–2026, 2019.

[22] Varatharajan R., Vasanth K., Gunasekaran M., Priyan M., Gao X.Z.: An adaptive decision based kriging interpolation algorithm for the removal of high density salt and pepper noise in images, *Computers & Electrical Engineering*, vol. 70, pp. 447–461, 2018.

[23] Wang Q., Ma J., Yu S., Tan L.: Noise detection and image denoising based on fractional calculus, *Chaos, Solitons & Fractals*, vol. 131, p. 109463, 2020.

[24] Wang S., Wang C., Xu C.: An image encryption algorithm based on a hidden attractor chaos system and the Knuth–Durstenfeld algorithm, *Optics and Lasers in Engineering*, vol. 128, p. 105995, 2020.

[25] Yahya A.A., Tan J., Su B., Hu M., Wang Y., Liu K., Hadi A.N.: BM3D image denoising algorithm based on an adaptive filtering, *Multimedia Tools and Applications*, vol. 79(27), pp. 20391–20427, 2020.

[26] Yang C.H., Chien Y.S.: FPGA implementation and design of a hybrid chaos-AES color image encryption algorithm, *Symmetry*, vol. 12(2), p. 189, 2020.

[27] Zhang Y., Chen A., Tang Y., Dang J., Wang G.: Plaintext-related image encryption algorithm based on perceptron-like network, *Information Sciences*, vol. 526, pp. 180–202, 2020.

Affiliations

Sunil B. Hebbale
KLE College of Engineering and TechnologyChikodi, India, sunilhebbale123@gmail.com

Dr. V.S. Giridhar Akula
Professor in CSE and Principal, KORM College of Engineering, Tadigotla, KADAPA, Andhra Pradesh,India, giridharakula456@gmail.com

Dr. Parashuram Baraki
Professor, Department of CS&E Smt. Kamala and Sri.Venkappa M. Agadi College of Engineering and Technology, Lakshmeshwar, Dist. Gadag, Karnataka, India, parashurambaraki456@gmail.com

Received: 27.09.2021
Revised: 29.12.2021
Accepted: 04.01.2022