Scalable Computing: Practice and Experience, ISSN 1895-1767, http://www.scpe.org
© 2020 SCPE. Volume 21, Issue 2, pp. 217–232, DOI 10.12694:/scpe.v21i2.1664

CHICKEN-MOTH SEARCH OPTIMIZATION-BASED DEEP CONVOLUTIONAL NEURAL NETWORK FOR IMAGE STEGANOGRAPHY

RESHMA V K* VINOD KUMAR. R. S† SHAHI D‡ AND SHYJITH M.B§

Abstract. Image steganography is considered as one of the promising and popular techniques utilized to maintain the confidentiality of the secret message that is embedded in an image. Even though there are various techniques available in the previous works, an approach providing better results is still the challenge. Therefore, an effective pixel prediction based on image stegonography is developed, which employs error dependent Deep Convolutional Neural Network (DCNN) classifier for pixel identification. Here, the best pixels are identified from the medical image based on DCNN classifier using pixel features, like texture, wavelet energy, Gabor, scattering features, and so on. The DCNN is optimally trained using Chicken-Moth search optimization (CMSO). The CMSO is designed by integrating Chicken Swarm Optimization (CSO) and Moth Search Optimization (MSO) algorithm based on limited error. Subsequently, the Tetrolet transform is fed to the predicted pixel for the embedding process. At last, the inverse tetrolet transform is used for extracting the secret message from an embedded image. The experimentation is carried out using BRATS dataset, and the performance of image stegonography based on CMSO-DCNN+tetrolet is evaluated based on correlation coefficient, Structural Similarity Index, and Peak Signal to Noise Ratio, which attained 0.85, 46.981dB, and 0.6388, for the image with noise.

Key words: Image steganography, Deep Convolutional Neural Network, Tetrolet transform, Chicken swarm optimization, Moth search optimization algorithm.

AMS subject classifications. 68T05

1. Introduction.

Nowadays, steganography plays a very significant role in various areas, which is utilized for the protection of data from unauthorized access. Steganography becomes very popular in secret message communication. In the steganography process, the sender must choose a suitable message, and the effective information needs to be hidden [38, 39, 40]. The object employed for carrying the message is known as the carrier image or the message carrier. The secret message is nothing, but the image, which is to be embedded in the carrier image. Stego-image is employed for carrying the hidden message. Hence, given the secret and the carrier image, the main aim of steganography is to build the stego-image for carrying the hidden message [9]. Steganography is the security approach for hiding the secret data. It has worked on concealing the image, text, video, or audio (sensitive data) inside other images, video or audio, text (cover) [10]. Steganography approaches are divided into video, text, protocol, Deoxyribonucleic Acid (DNA), and video steganography.

Image steganography is the technique for hiding the unnatural hidden message in carrier image, so the quality of carrier image has a small change, and hence no one can find it [11]. In the current, most of the steganographic approaches are employed for reducing the distortion function correlated with the statistical detectability [13]. Image steganography is defined by hiding the secret message in the carrier image so that the receiver can recover the watermark message when the warder does not detect the secret information. Most of the image steganography approaches attained their goals by embedding process that leaves evidence for distortion [8]. The image-based stego is considered as the effective cover medium due to its easy human remembrance possibility, and popularity [10]. Image steganography is also utilized for various applications, like safe communication between two parties [12, 14], captioning and contents protection [12, 16, 17], securing

*1Research Scholar in Noorul Islam Centre For Higher Education, Kumaracoil, Tamil Nadu, and Assistant Professor, Department of CSE, Jawaharlal College of Engineering and Technology, Palakkad, Kerala (reshmavk14@gmail.com).
†Head of Department, Electronics and Communication Engineering, Noorul Islam Centre for Higher Education, Kumaracoil, Tamil Nadu
‡Assistant Professor, Dept. of ECE, Noorul Islam Centre for Higher Education, Kumaracoil, Tamil Nadu
§Assistant Professor, Dept of CSE, Jawaharlal College of Engineering and Technology, Palakkad, Kerala

online voting systems, secure mobile computing [12, 15], privacy-protection of medical records, personalized secure image retrieval, and secure surveillance systems [12, 18].

Image Steganography allows two parties for secure communication. Image steganography approaches are divided into two, namely, frequency, and spatial domain [19]. The process of the spatial domain is to allow whole pixels of carrier image, and the embedding of the secret message into the frequency domain is performed after numerous conversions for converting the image to the frequency domain. The name "stego" comes from a concealed secret message in the image. Several methods are related to the spatial domain, like Gray level modification (GLV), Least Significant Bit (LSB), and so on [20]. The frequency domain-based methods are Discrete Fourier Transform (DFT), and Discrete cosine transform (DCT), and so on [11].

In this research work, the image stegonography is performed using the proposed pixel-prediction approach. The developed pixel prediction-based approach employs three phases for hiding the watermark information in the image as, identification, embedding, and extraction phase. Initially, the input image is subjected to the pixel prediction phase in which the appropriate pixels are extracted from the image. The features, like edge information, pixel coverage, texture features, wavelet energy, Gabor feature value, and scattering values are extracted from the pixels using DCNN classifier, which is trained by CSO and MSO based on error. In the third phase, the watermark message is embedded in the input image using tetrolet transform that assures the embedding strength. Finally, the extraction of a watermark image is done by applying Tetrolet Transform (TT).

The main contribution of the research paper is enlisted below:

- The pixel identification is carried out based on the DCNN classifier using error, which is trained by applying CSO and MSO for finding the effective pixel employed for the embedding process.
- The features are extracted effectively from the pixel for generating the prediction map, which is followed with the embedding procedure to embed the watermark message in the input image using TT. The inverse TT is employed to extract the secret message from the input image.

The paper is structured as follows: Section 2 discusses existing methods of image stegonography with challenges of the methods. The proposed method of CMSO-DCNN+tetrolet is demonstrated in section 3, and section 4 provides the results and discussion. At last, section 5 concludes the research work.

2. Literature Survey. This section presents the literature survey of several methods utilized for the image stegonography and the challenges of the existing works.

Several methods related to image stegonography are described, and analyzed as follows: Yuileong Yeung et al. [20] developed binary image steganography to reduce the flipping distortion on Local Texture Pattern (LTP) and designed flexible carriers of syndrome-trellis code (STC). Here, the security for both vision and statistics was found better, but it failed to access the scalable and nearly continuous capacity upper bound. Adnan Gutub and Maimoona Al-Ghamdi [10] presented counting-based secret sharing method to improve the shares reconstruction and distribution. The method failed to define the maximum and minimum shares for reconstructing the secret. Weixuan Tang [21] employed a Convolutional Neural Network based on adversarial embedding (ADV-EMB) for image stegonography. Here, the security was found better, but failed to detect the stego images in the current iteration. Soumendu Chakraborty et al. [22] developed Predictive Edge Adaptive image steganography in which the selected area of the cover image was determined based on Modified Median Edge Detector (MMED) to embed the binary payload (data). The developed method achieved limited level of distortion and best embedding rate. More bits were needed in sharper edges for adaptive selection.

Dipti Kapoor Sarmah and Anand J. Kulkarni [5] presented Multi Random Start Local Search (MRSLS) for achieving a better balance between secret text capacity, security, and image quality. This framework employs Cohort Intelligence (CI) for attaining the enhanced quality in the image. The method failed to consider the cohort intelligence approach was integrated with a cuckoo search algorithm for enhancing efficiency. Aref Miri and Karim Faez [23] developed an integer wavelet transform for mapping cover images with a specific frequency domain. The Most Significant Bit (MSB) bit was utilized for categorizing the edge coefficients in the frequency domain. Embedding the secret bits in the frequency coefficients are needed for obtaining the stego image. S.I. Nipanikar and V. Hima Deepthi [6] presented edge and wavelet transformation approach to finding the accurate location to embed the message. Here, the edges are detected accurately based on the wavelet coefficient, and the intensity of pixel, but does not consider another optimization algorithm for improving the cost estimation of the pixel. Tomas Denemark and Jessica Fridrich [7] developed an approach for inferring the accurate direction

of changes made in steganographic embedding. This change was incorporated with cost-based steganography for reducing the embedding costs based on the multiplicative modulation factor, but more than two acquisitions were needed for the embedding process.

2.1. Challenges. This section deals with the challenges faced by the existing techniques of image stegonography.

- Once the data embedding is done, the changes in the image statistics are predicted using the steganalyzers, and the bit planes employed in the typical image are less correlated, which is one of the challenges faced by the image stegonography [24].
- Other challenges face by image steganography are the visibility of extracted images at the receiver side by changing the value of the blending coefficient is very sensitive to modification [8].
- The stego image quality and the capacity of embedding is the challenging task for designing the secure binary image by enhancing the undetectability [14].
- In real-time, if there is additive noise in the cover image, there is the possibility for the poor PSNR or in other words, the image quality is affected.
- The embedding capacity associated with the steganography is not significant using the color parity, which depends on the order of the colors, and the capacity of the image is very less, which is the major challenge in embedding the stego images from the cover image [16].

3. Image Stegnography using the proposed Chicken-Moth search optimization-based deep convolutional neural network. This section presents the Image Steganography using Chicken-Moth search optimization algorithm. Figure 3.1 deliberates the block diagram of the proposed Chicken-Moth search optimization algorithm for Image Steganography. At first, the input medical image is given to the pixel prediction phase in which the features, like wavelet, pixel coverage, edge details, and so on will be extracted from the image, which is subjected to the pixel identification using the proposed CMSO-based DCNN. Thus, the proposed CMSO algorithm is developed by integrating CSO [37] with the MSO [36]. Thus, the pixels to be embedded will be decided using the Deep CNN that will be trained using the CMSO algorithm. Here, the embedding of the secret message is done by applying TT such that it offers fast and efficient image representation. The extraction of the secret image is done at last by applying the inverse transform.

Let us consider the input medical image M with dimension $X \times Y$. Here, the term U represents the watermark message of dimension $P \times Q$. The watermark image is in the form of binary. The representation of the input, and embedded watermark image is given by

$$M = \{m_{uv}\}; 1 \leq u \leq X; 1 \leq v \leq Y \tag{3.1}$$

$$U = \{S_{xy}\}; 1 \leq x \leq P; 1 \leq y \leq Q \tag{3.2}$$

where, the term m_{uv} refer to the input image pixels, ranging between 0 to 255, and S_{xy} denotes the watermarked image pixels, having the binary value as 0 or 1.

3.1. Pixel prediction. Extraction of features. In the pixel prediction phase, the feature extraction is performed based on seven features, such as wavelet energy, edge information, scattering value, Gabor filter, pixel coverage, Local binary pattern (LBP), and Tetrolet transform. The feature extraction step carried out in this paper is explained as follows.

3.1.1. Wavelet energy. The TT [25] is subjected to the input medical image to achieve the wavelet energy. The wavelet transform is employed for decomposing the original image to four sub-bands, which includes HL, HH, LL, and, LH. The four bands are represented as, $\{L_1, L_2, L_3, L_4\}$. The tetrolet energy for the pixel is expressed as

$$TT(m_{uv}) = \{L_1, L_2, L_3, L_4\} \tag{3.3}$$

where $TT(m_{uv})$ represents the tetrolet function. In this paper, the embedding phase focussed on HL band, hence the energy consumed in HL band is taken as the tetrolet energy feature and is represented as $F_1 = L_3(m_{uv})$.

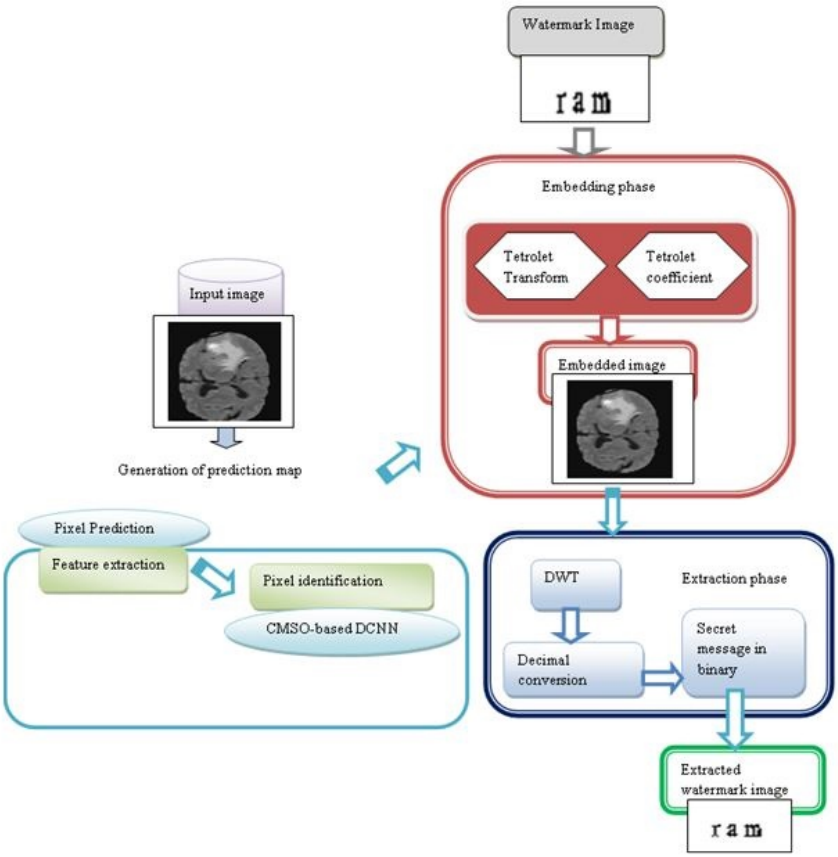

FIG. 3.1. *Schematic diagram of image stegonography using the proposed CMSO-based DCNN*

3.1.2. Pixel coverage. The pixel coverage is computed based on the mean value of neighborhood pixel to provide the information about the coverage value of each pixel. Assume the pixel m_{uv} in image M with N number of neighboring pixels, and the pixel coverage is expressed as,

$$F_2 = \frac{1}{N} \sum_{a=0}^{N-1} m_{uv}^a \tag{3.4}$$

where m_{uv}^a is the symbol indicates the a^{th} neighbor pixel of $m_{uv}$and N denotes the neighboring pixel.

3.1.3. Edge detection. It is the process of determining whether the pixel is available in the corner edge or not. In edge information feature, if the pixel is in the edge then the value is fixed as one or otherwise the value is set to zero. The edge information for the pixel is given by

$$g(u, v) = A(m_{uv}) \tag{3.5}$$

where $A(m_{uv})$ refer to the edge information, and the output of the edge information is represented as $F_3 = g(u, v)$.

3.1.4. Scattering value. The scattering transform [26] is employed for obtaining scattering coefficients. This transform finds the texture information by applying the filter convolution in the pixel. The scattering coefficient is given by

$$K[M] = |||M \otimes \eta_{b1}| \otimes \eta_{b2}|K| \otimes \eta_{bh}| \otimes A(e)| \tag{3.6}$$

where the average filter is denoted as $A(e)$, and the term indicates the filter banks. The output obtained from the scattering transform is denoted as $F_4 = K(m_{uv})$.

3.1.5. Gabor feature. The Gabor filter [27] is employed for identifying the time-frequency location of the pixel and to achieve the robust against various brightness or contrast of the image. For the feature extraction, 2D Gabor filter is the broadly utilized filter and the filter function is given by,

$$P(u, v, \varphi, h, \chi) = \frac{1}{2\pi\sigma^2} exp\left(-\frac{u^2 + V^2}{2\sigma^2}\right) * exp\{2\pi u_m(hu\cos\theta + hv\sin\theta)\} \tag{3.7}$$

where $P(u, v, \varphi, h, \chi)$ denote the Gabor filters with pixel. The term $u - m$ denotes the imaginary part and the value is $\sqrt{-1}$. The sinusoidal wave frequency in the Gaussian filter is represented as h . The output of the Gabor filter is denoted as F_5, and is expressed as$D(\{u, v\} = P(m_{uv})$. Therefore, $F_5 = D\{u, v\}$.

3.1.6. Local Binary Pattern. The crisp form of LBP [28] utilizes the neighbourhood pixel properties to explain each pixel. It is more resistant, efficient, and simple to make changes in gray-level using lighting variations. The obtained feature is indicated as, $F_6 = V(u, v)$. The texture features obtained from the extraction phase is expressed by

$$V(u, v) = LBP)m_{uv} = \sum_{a=0}^{N-1} r(t_a - t_f)2^n \tag{3.8}$$

where the term $t_f$and t_a denotes the centre and neighbor pixels gray value. Then, the membership function is calculated as

$$r(b) = \left\{ \begin{array}{ll} 1; & b \geq 0 \\ 0; & \text{otherwise} \end{array} \right\} \tag{3.9}$$

3.1.7. Tetrolet Transform. The tetrolet descriptor [25] is an adaptive Haar wavelet transform, to support tetrominoes, which is formed by joining four squares with similar size. Here, the input low-pass image is divided into blocks and local tetrolet basis are generated based on the geometry of the image. The steps involved in the tetrolet transform are illustrated below.

i) *Initialization*: The input image is split into blocks of size.

ii) *Representation of image blocks as the sparsest tetrolet*: Each of the image blocks is subjected to the sparsest tetrolet representation and for every individual blocks, a total of 117 tetromino coverings are admitted each of which is given to the Haar wavelet transform along with four low pass coefficients for generating 12 Tetrolet coefficients. For the individual block, the tetrolet decomposition is done at the optimum based on 12 tetrolet coefficients to obtain the final sparse image.

iii) Representation of the high pass and Low pass coefficients: The steps involved in the Tetrolet decomposition algorithm is preceded with the arrangement of matrix based on reshape function.

iv) Tetrolet Coefficients: After representing the sparse matrix for the individual blocks, the high pass as well as the low pass matrices is kept safe for the usage of future.

v) Termination: The steps (ii) to (iii) are repeated for the low pass image and the output obtained in the binary image, which is denoted as F_7. Therefore, the extracted features are represented as

$$J^{red} = \{F_1, F_2, F_3, F_4, F_5, F_6, F_7\} \tag{3.10}$$

The size of the extracted features is denoted as $[1 \times 7]$.

3.2. Pixel identification using DCNN. Once the features are extracted, DCNN is utilized for pixel identification. The DCNN is utilized for generating the prediction map for image pixels. The DCNN classifier [29] uses the extracted features J^{red} as input and generates the prediction map based on input image. The architecture of DCNN and the algorithmic steps of the CMSO-based DCNN are described bellow.

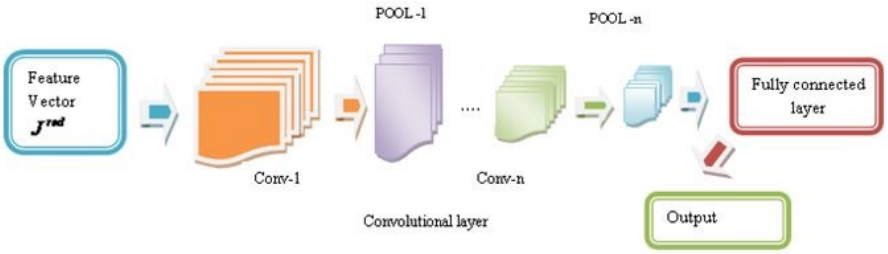

FIG. 3.2. *Architecture of DCNN for the construction of prediction map*

3.2.1. Architecture of the DCNN. The basic architecture of the DCNN [29] is discussed in this section with its architecture in figure 3.2. The DCNN comprises of the number of convolutional (conv) layers, pooling (POOL). The architecture of the Deep CNN [34,35] is deliberated in figure 3.2 and the architecture of DCNN consists of three layers, such as pooling (POOL), convolutional (conv), and Full Connected (FC) layers. Among the three layers of DCNN, each of the layers constitutes specific function. The main function of the conv layers is to generate the feature maps from the segments of the pre-processed image and these feature maps are further sub-sampled down in the pool layers. The third layer is the FC layer, where the classification is progressed. The convolutional layer engages in mapping the input such that the input maps undergo convolution with the convolutional kernels in order to develop the output map The size of the output map is similar to the kernel number, and the size of the kernel matrix is $[3 \times 3]$. Thus, making it clear that the conv layers is the multilayer loop of input maps, kernel weights, and output maps. In the first conv layer, there are a number of inputs and outputs, whose size reduces in the successive conv layers such that the classification accuracy of the objects depends on the number of the layers in the DCNN.

Conv layers: The responsibility of the conv layers relied on obtaining the patterns buried in the input feature vector using the conv filters that are connected using the receptive fields, which act as an interconnection between the neurons in the previous layer with the successive conv layers through the trainable weights. The feature maps are developed through the convolution of the input feature vector with the trainable weights in such a way that the trainable weights are derived using the hybrid optimization algorithm. The neurons of the single layer engage themselves in extracting the variable features available in various location based on the variable weights of the single layer. Let us assume the input to the deep CNN is G and hence, the output from the conv layer is given as

$$(G_c^b)_{i,j} = (B_c^b)_{i,j} + \sum_{d=1}^{\varpi_1^{d-1}} \sum_{k=\varpi_1^1}^{\varpi_1^1} \sum_{n=\varpi_1^1}^{\varpi_2^1} (\vartheta_{c,d}^b)_{k,n} * (J^{red})_{i+k,j+n} \tag{3.11}$$

where the symbol $*$ refers to the convolutional operator that paves way for obtaining the local patterns from the alternative conv layers, J^{red} is the extracted features, $(B_c^b)_{i,j}$ indicates the fixed feature map or the output from the b^{th} conv layers centered as (i, j). The output from the previous $(b-1)^{th}$layer forms the input to the l^{th} conv layer. Let the weights of the conv layers be denoted as,$\vartheta_{c,d}^b$, which is the weights of b^{th} conv layer and the bias of b^{th} conv layer is denoted as (B_c^b). Let us consider d, k and n as the notations of feature maps.

ReLU layer: ReLU is abbreviated as Rectified Linear Unit that applies non-saturating activation function. It eliminates the negative values effectively from the activation map by fixing them to zero, also improves the nonlinear properties of decision function without affecting the receptive fields of the convolution layer. The neurons in conv layers are arranged in 3-dimensions along the depth, height and width, so as for extracting the features from all the dimensions of ReLU layer, which uses an element-wise activation function to simplify the computation using the removal of negative values. The output from the l^{th} layer is the activation function of the preceding $(k-1)^{th}$ layer, and is expressed as

$$G_c^b = Afn(G_c^{b-1}) \tag{3.12}$$

The importance of ReLU layer is regarding the speed of DCNN, which is enhanced and offers the ability to deal with large number of networks.

***POOL layers*:** It is a non-parametric layer with no weights, and bias, undergoing a fixed operation. The importance of POOL layer is to mitigate the spatial dimensions of the input and minimizes the computational complexity.

***FC layers*:** The patterns generated using the pooling and the conv layers form the input to the fully connected layers that are subjected to high-level reasoning. The output from the fully-connected layers is given as

$$(H_c^b) = \delta(G_c^{b-1}) with (G_c^b) = \sum_{d=1}^{\varpi_1^{d-1}} \sum_{k=\varpi_1^1}^{\varpi_1^1} \sum_{n=\varpi_1^1}^{\varpi_2^1} (\vartheta_{c,d}^l)_{k,n} * (J^{red})_{i+k,j+n} \tag{3.13}$$

where $(\vartheta_{c,d}^l)_{k,n}$ denotes the weight.

3.2.2. Training of DCNN based on Chicken-Moth search optimization. The proposed CMSO is the integration of CSO algorithm and MSO. The CSO [37] is a bio-inspired optimization approach, which mimics the hierarchy of chickens swarm and the behavior of searching the food, in which each chicken denotes the potential solution for optimization issue. The hierarchical order is very important in social lives of chicken. Here, the chicken swarms are categorized into many hens, one rooster, and chicks. For each group, the chicken finds the rooster, hen, and chick based on fitness value of chicken. MSO [36] is an advanced meta-heuristic algorithm motivated with photo axis and levy flights of the moths. MS algorithm can search the best solution effectively with improved accuracy. Moreover, the algorithm negotiates complex operations, and thus, the execution of MS algorithm is easy and flexible. The steps involved in the proposed CMSO are described as follows.

a) Initialization: In the first step, the position of the moths is randomly initialized, represented as $\{X_{ef}, 1 \leq e \leq t; 1 \leq f \leq p\}$ where t is the population size, and p denotes the dimension. $X \in \{(\vartheta_{c,d}^l, B_c^b, \vartheta_{c,d}^b)\}$.

b) Evaluation of the objective function: The selection of the optimal location of the chicken is performed based on minimization problem. The minimal value of the objective function describes the better solution and therefore, the solution with the minimum value of the error is chosen as the best solution. The error is determined as

$$MSE = \frac{1}{X}[\sum_{h=1}^{X} H_{target} - H_c^b] \tag{3.14}$$

where H_{target}, H_c^b and are the estimated and target output of the classifier. The term X denotes the total number of samples.

c) Location update using levy flights: After evaluating the objective function, the solution undergoes position update based on the levy flight update, and it is mentioned as follows:

$$X_{e,f}^{\tau+1} = X_{e,f}^{\tau} + \varepsilon.F(z) \tag{3.15}$$

Rearranging the above equation,

$$X_{e,f}^{\tau} = X_{e,f}^{\tau+1} - \varepsilon.F(z) \tag{3.16}$$

where $X_{e,f}^{\tau}$ specifies the location of the moth at the iteration τ , and the term $F(Z)$ signifies to the step drawn due to the movement of levy flight. The parameter ε indicates the scaling factor and is expressed as,

$$\varepsilon = \frac{W_{max}}{\tau^2} \tag{3.17}$$

where W_{max} refer to the maximum step walk. Then, the levy distribution $H(t)$ is represented as

$$H(t) = \frac{(\alpha-1)\lceil(\alpha-1)\sin(\frac{\pi(\alpha-1)}{2})}{\pi a^2} \tag{3.18}$$

where q is greater than 0.$\lceil(y)$ is the gamma function.

d) Fly straightly: The location of the moth is also influenced by light source, and the upgrade solution is represented as follows:

$$X_{e,f}^{\tau+1} = \lambda \times (X_{e,f}^{\tau} + \beta.(X_{best}^{\tau} - X_{e,f}^{\tau})) \tag{3.19}$$

where X_{best}^{τ} denotes the best location of the moth, and the term β signifies the acceleration factor. The scaling factor is represented as λ. During the fly straightly movement, the location of the moth is influenced by the location of the light source. Here, the acceleration constant influences the convergence speed of algorithm. In some cases, the position of the moth goes beyond the position of the light source. Then, the equation (16) is modified with the CSO for enhancing the effectiveness of the approach and to find solutions to various optimization issues. The standard equation of the movement of chick is given by

$$X_{e,f}^{\tau+1} = X_{e,f}^{\tau} + AB * (X_{w,f}^{\tau} - X_{e,f}^{\tau}) \tag{3.20}$$

$$X_{e,f}^{\tau+1} = X_{e,f}^{\tau}[1 - AB] + AB * X_{w,f}^{\tau} \tag{3.21}$$

Substituting equation (16) in equation (21),

$$X_{e,f}^{\tau+1} = [X_{e,f}^{\tau+1} - \varepsilon.F(z)][1 - AB] + AB * X_{w,f}^{\tau} \tag{3.22}$$

$$X_{e,f}^{\tau+1} - [1 - AB]X_{e,f}^{\tau+1} = (AB - 1)\varepsilon.F(z) + AB * X_{w,f}^{\tau} \tag{3.23}$$

$$X_{e,f}^{\tau+1} = \frac{1}{AB}[(AB - 1)\varepsilon.F(z) + AB * X_{w,f}^{\tau}] \tag{3.24}$$

Thus by equation (24), the position update of the moths can be obtained, using the location of the moths in its preceding iteration, light absorption coefficient, attractiveness, and the distance between the moths.

e) Finding the best solution: The feasibility of the solution is computed based on the objective function. If the newly generated solution is best than the previous one, then it is changed by the new solution.

f) Termination: After a certain iteration limit, the algorithm terminates, and the optimal solution is retained at the end of the procedure. Thus, the best solution, chosen using the proposed CMSO algorithm is utilized for embedding the secret message. The predicted map is denoted as O.

3.2.3. Embedding using Tetrolet transform. This section presents the embedding phase using Tetrolet transform for image steganography. The embedding is utilized for hiding the secret message in the HL band, and the tetrolet transform is given for extraction and embedding process. During embedding the input image is divided into sub-bands and the watermark is embedded using tetrolet coefficient. The embedding process of pixel prediction approach is depicted in figure 3.3.

At first, the sub-bands of the input medical image are generated based on tetrolet transform in the embedding phase. The tetrolet transform is employed for acquiring tetrolet coefficient. When tetrolet transform is applied to input medical image, four bands are generated, and is expressed as,

$$TT(M) = \{L_1, L_2, L_3, L_4\} \tag{3.25}$$

In the tetrolet coefficient, each band refers to frequency and energy. Embedding the watermark image is performed in band L_3 is denoted as

$$L_3^* = L_3 + U^*\gamma^*O \tag{3.26}$$

where the term U denotes the watermark message, and L_3 is the symbol of HL band. The term represents the HL band embedded by the watermark message, and γ be the embedding strength. The secret message U is embedded in L_3using the predicted map and the embedding strength from the classifier. The predicted map in the classifier determines the appropriate pixels and the intensity is defined using the embedding strength. The watermark image embedded to the HL band is expressed as, $\{L_1, L_2, L_3^*, L_4\}$. After the embedded of watermark message, the inverse tetrolet transform is applied to obtain the embedded image. The embedded input medical image is expressed as

$$M^* = ITT\{L_1, L_2, L_3^*, L_4\} \tag{3.27}$$

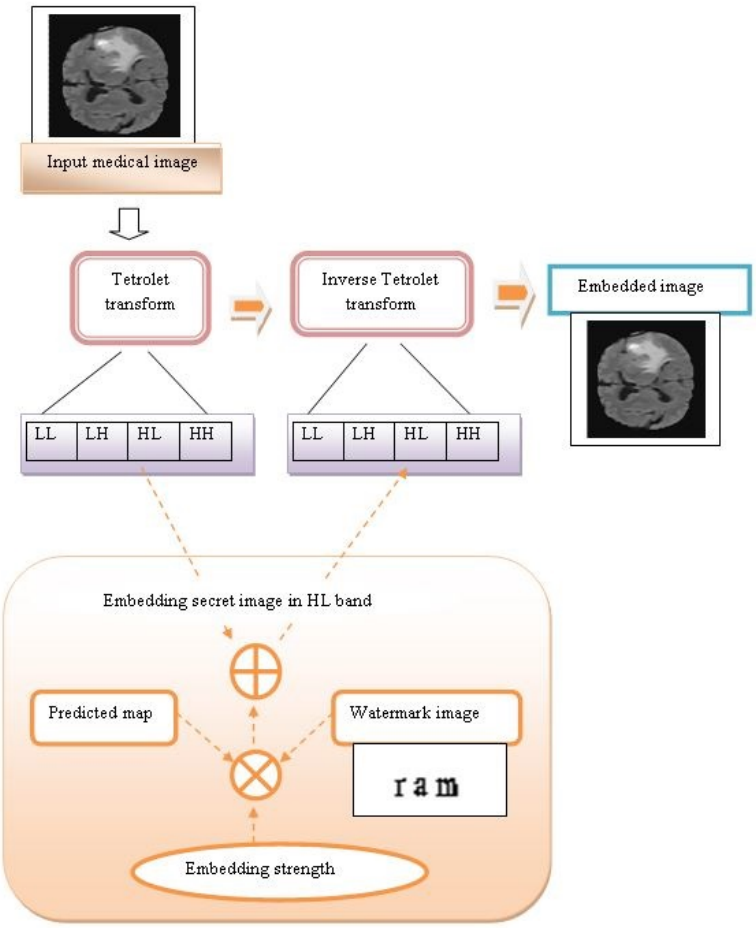

FIG. 3.3. *Embedding process of the secret image*

3.2.4. Watermark message retrieval. Once the embedded image is determined, then, the image is transmitted to the extraction phase. The TT is applied for obtaining tetrolet coefficients. At the extraction steps, the receiver extracts the watermark message. The extraction of secret message is expressed as

$$TT(M^*) = \{N_1^{ex}, N_2^{ex}, N_3^{ex*}, N_4^{ex}\} \tag{3.28}$$

where N_1^{ex}, N_2^{ex} , N_3^{ex*} and N_4^{ex} refers to the four bands obtained from the tetrolet process. The extraction process is formulated as,

$$T^{ex} = L_3^{Ex*} - L_3 \tag{3.29}$$

In the extraction phase, the watermarked message is represented as $U^{ex} = \{U_{xy}^{ex}\}$, which is utilized for stegonography.

4. Results and Discussion. The results and discussion of the developed CMSO-DCNN+tetrolet for image stegonography are demonstrated in this section with an effective comparative analysis to prove the effectiveness of proposed method.

4.1. Experimental Setup. The experimentation of image stegonography method is performed in system with 2 GB RAM, Intel i-3 core processor, Windows 10 Operating System. The proposed method is executed in MATLAB.

4.2. Database description. The dataset is taken from the BRATS database [30] for image stegonography. Here, the image of every patient is collected as four modalities, like T1, T1C, FLAIR, and T2. In this dataset, all the datasets are manually segmented, by one to four rates, which follow the similar annotation protocol, approved by experienced doctors.

4.3. Performance metrics. The evaluation of the developed model is performed based on three metrics namely, Correlation factor, SSIM, and PSNR.

a) PSNR: The quality of frame is determined using PSNR. The maximum value of PSNR assures that the system is better and it is represented in decibel (dB).

$$PSNR = 10\log_{10}(\frac{m_{max}^2}{MSE}) \tag{4.1}$$

where the term m_{max} indicates the highest pixel value for M^{th} image.

b) SSIM index: For predicting the perceived quality of the video frame, SSIM is used. The SSIM value is maximal for the effective method. Here, the SSIM is measured by two windows, such as χ_1 , and χ_2 .

$$SSIM(\chi_1, \chi_2) = \frac{(2\eta_{\chi_1}\eta_{\tau_2} + \varphi_1)(2\kappa_{\chi_1,\chi_2} + \varphi_2)}{(\eta_{\chi_1}^2 + \eta_{\chi_2}^2 + \varphi_1)(\kappa_{\chi_1}^2 + \kappa_{\chi_2}^2 + \varphi_2)} \tag{4.2}$$

where $\eta_{\chi_1}$and η_{χ_2} represents the mean value of pixels for two windows, and the variance of pixels are denoted as κ_{χ_1} and κ_{χ_2} denotes the variance of pixels. The terms $\varphi_1$and φ_2 are utilized for stabilization.

c) Correlation factor: The correlation coefficient offers statistical relationship among the original image and embedded video image.

$$CF(\chi_1, \chi_2) = \frac{Cov(\chi_1, \chi_2)}{\kappa_{\chi_1}, \kappa_{\chi_2}} \tag{4.3}$$

where the term $Cov(\chi_1, \chi_2)$represents the covariance factor.

4.4. Experimental Results. The experimental results obtained by the developed technique are discussed in this section. Figure 4.1 depicts the experimental results obtained from the proposed method without using noise in the image, and salt and pepper noise, impulse noise, and Gaussian noise added in the image. Figure 4.1 a) depicts the input image, and figure 4.1 b) depicts the watermark message. The embedded image is shown in figure 4.1 c), and figure 4.1 d) depicts the final extracted message.

4.5. Comparative techniques. The methods, such as random [31], sequential [32], optimal order [33], SVNN-wavelet [34], Cost Function for Image Steganography Using Wavelet (CWSM) [35], SVNN-Contourlet, Moth+tetrolet, DCNN+Contourlet, and are used for the comparison with the proposed CMSO-DCNN+tetrolet for the analysis.

4.6. Comparative analysis.

4.6.1. Analysis using image without noise. The comparative analysis of the developed method is analyzed based on correlation coefficient, PSNR, and SSIM without adding noise in the image is shown in figure 4.2. Figure 4.2 a) illustrates the analysis based on correlation coefficient by varying the number of images. When the number of image is 1, then the corresponding correlation coefficient values computed by existing random, sequential, optimal order, CWSM, SVNN-Wavelet, SVNN-Contourlet, Moth+tetrolet, DCNN+Contourlet, and the proposed CMSO-DCNN+tetrolet are found to be 0.93, 0.98, 0.98, 0.98, 0.98, 0.98, 0.978, 0.98, and 0.98, respectively. The comparative analysis based on PSNR is depicted in figure 4.2 b). For image 2, the PSNR values achieved by random, sequential, optimal order, CWSM, SVNN-Wavelet, SVNN-Contourlet, Moth+tetrolet, and DCNN+Contourlet, and the proposed model are 39.15dB, 39.533dB, 40.08dB, 40.02dB,42.81dB, 42.81dB, 46.43dB,46.71dB, and 47dB, respectively. The analysis in terms of SSIM is depicted in figure 4.2 c). For image 3, the existing techniques, like random, sequential, optimal order, CWSM, SVNN-Wavelet, SVNN-Contourlet, Moth+tetrolet, and DCNN+Contourlet, possesses the SSIM of 0.484, 0.497, 0.959, 0.959, 0.959, 0.96, 0.956, and 0.952, respectively, which is comparatively lower than the CMSO-DCNN+tetrolet. For the same image, the developed CMSO-DCNN+tetrolet acquired the SSIM of 0.96.

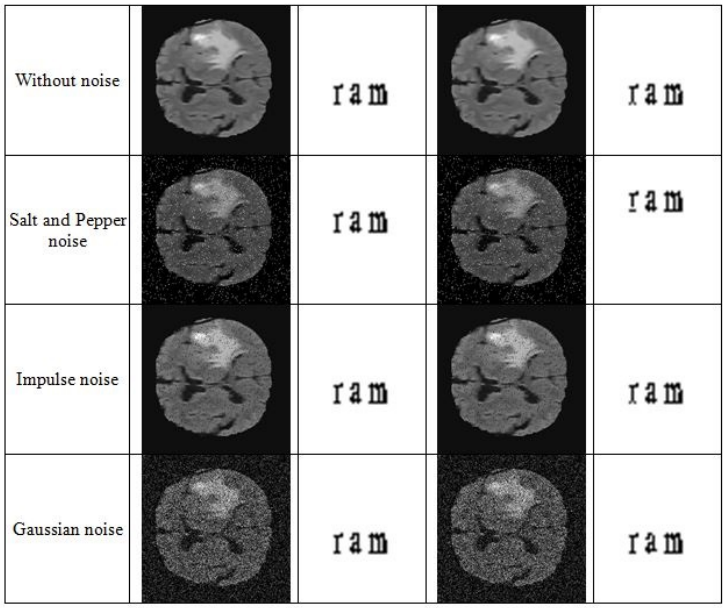

FIG. 4.1. *Sample results a) Input image b) Watermark message c) Embedded image, and d) Final extracted message.*

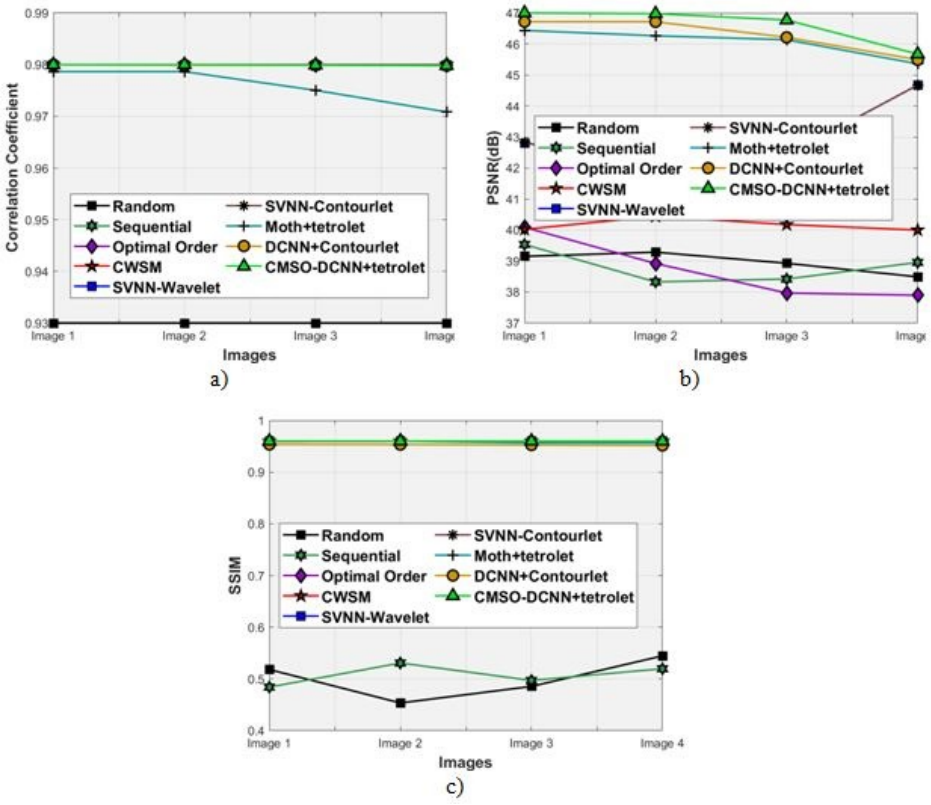

FIG. 4.2. *Comparative analysis using image without noise (a) Correlation coefficient, b) PSNR, and (c) SSIM*

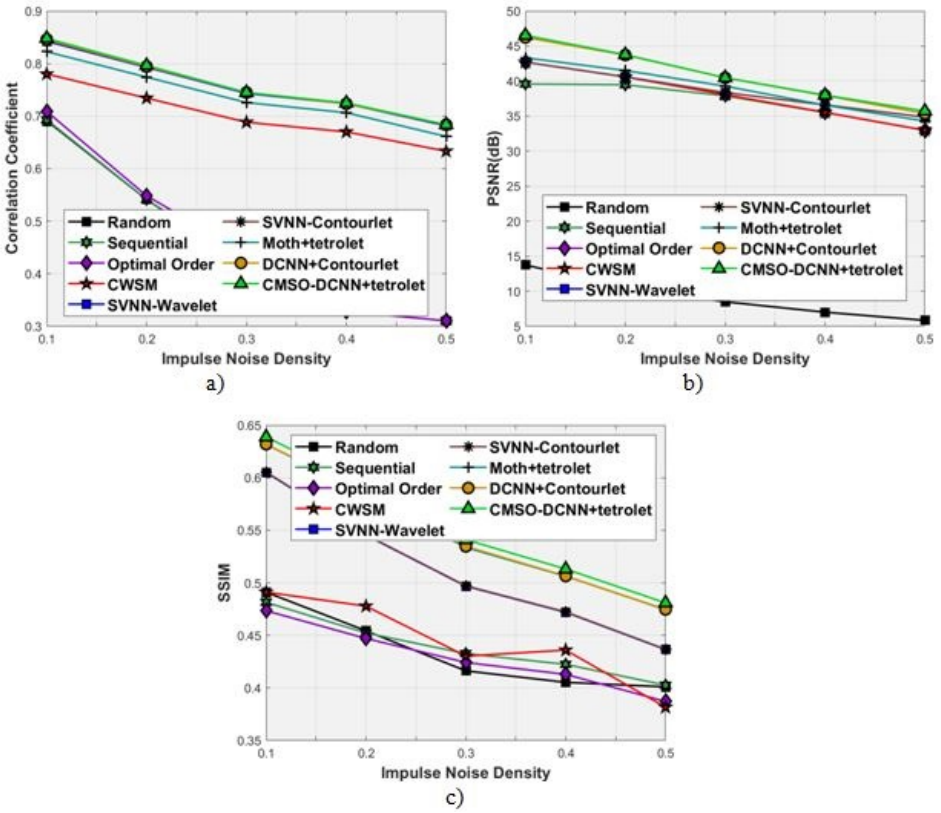

FIG. 4.3. *Comparative analysis by adding impulse noise (a) Correlation coefficient, b) PSNR, and (c) SSIM*

4.6.2. Analysis using image with Impulse noise. The comparative analysis of the developed method is analyzed based on correlation coefficient, PSNR, and SSIM with impulse noise is shown in figure 4.3. Figure 4.3 a) shows the analysis in terms of correlation coefficient by varying the impulse noise density. Similarly, when the impulse noise density is increased to 0.5, the methods, random, sequential, optimal order, CWSM, SVNN-Wavelet, SVNN-Contourlet, Moth+tetrolet, and DCNN+Contourlet, attained the correlation coefficient of 0.3105, 0.3105, 0.3105, 0.6333, 0.6816, 0.6826, 0.6613, and 0.6837, whereas the correlation coefficient of the developed method is 0.683. The comparative analysis based on PSNR is depicted in figure 4.3 b). When the impulse noise density=0.2, the PSNR values achieved by random, sequential, optimal order, CWSM, SVNN-Wavelet, SVNN-Contourlet, Moth+tetrolet, and DCNN+Contourlet, and the proposed model are 10.53dB, 39.48dB, 40.59dB, 40.59dB,40.601dB, 40.608dB, 41.512dB, 43.71dB, and 43.77dB, respectively. The analysis in terms of SSIM is depicted in figure 4.3 c). When the impulse noise density is 0.3, the existing techniques, like random, sequential, optimal order, CWSM, SVNN-Wavelet, SVNN-Contourlet, Moth+tetrolet, and DCNN+Contourlet, possesses the SSIM of 00.416, 0.432, 0.424, 0.430, 0.496, 0.497, 0.533, and 0.534, respectively, which is comparatively lower than the CMSO-DCNN+tetrolet. For the same impulse noise density, the developed CMSO-DCNN+tetrolet acquired the SSIM of 0.541.

4.6.3. Analysis using image with salt and pepper noise. The comparative analysis of the developed method is analyzed based on correlation coefficient, PSNR, and SSIM with salt and pepper noise is depicted in figure 4.4. Figure 4.4 a) illustrates the analysis based on correlation coefficient by varying the number of salt and pepper noise density. When the salt and pepper noise density=0.3 , the existing techniques, like random, sequential, optimal order, CWSM, SVNN-Wavelet, SVNN-Contourlet, Moth+tetrolet, DCNN+Contourlet, and the proposed CMSO-DCNN+tetrolet possesses the correlation coefficient of 0.4129, 0.4167, 0.4379, 0.688, 0.7434,

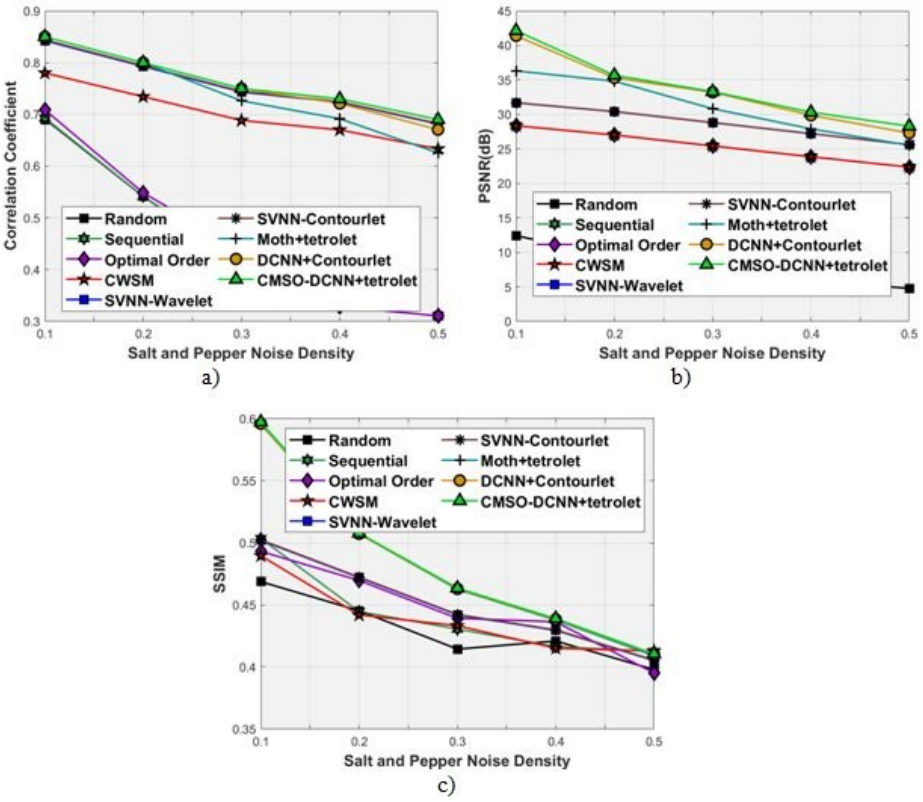

FIG. 4.4. *Comparative analysis by adding salt and pepper noise (a) Correlation coefficient, b) PSNR, and (c) SSIM*

0.7445, 0.726, 0.75, and 0.75, respectively. The comparative analysis based on PSNR is depicted in figure 4.4 b). When the density of salt and pepper noise is 0.3, the PSNR values achieved by random, sequential, optimal order, CWSM, SVNN-Wavelet, SVNN-Contourlet, Moth+tetrolet, and DCNN+Contourlet, and the proposed model are 7.308 dB, 25.42 dB, 25.429 dB, 25.432 dB, 28.806 dB, 28.824 dB, 30.830 dB, 33.211 dB, and 33.30 dB respectively. The analysis in terms of SSIM is depicted in figure 4.4 c). When the salt and pepper noise density is 0.4, the existing techniques, like random, sequential, optimal order, CWSM, SVNN-Wavelet, SVNN-Contourlet, Moth+tetrolet, and DCNN+Contourlet, possesses the SSIM of 0.421, 0.416, 0.436, 0.415, 0.429, 0.429, 0.437, and 0.438, respectively, which is comparatively lower than the CMSO-DCNN+tetrolet. For the same noise density, the developed CMSO-DCNN+tetrolet acquired the SSIM of 0.4389.

4.6.4. Analysis using image with Gaussian noise. The comparative analysis of the developed method is analyzed based on correlation coefficient, PSNR, and SSIM with Gaussian noise is depicted in figure 4.5. Figure 4.5 a) illustrates the analysis based on correlation coefficient by varying the Gaussian noise variance. When the Gaussian noise variance 0.4 is considered, the existing techniques, like random, sequential, optimal order, CWSM, SVNN-Wavelet, SVNN-Contourlet, Moth+tetrolet, DCNN+Contourlet, and the proposed CMSO-DCNN+tetrolet possesses the correlation coefficient of 0.328, 0.328, 0.328, 0.670, 0.7130, 0.7140, 0.7389, 0.7189, and 0.7201, respectively. The comparative analysis based on PSNR is depicted in figure 4.5 b). When the Gaussian noise variance is 0.2, the PSNR values achieved by random, sequential, optimal order, CWSM, SVNN-Wavelet, SVNN-Contourlet, Moth+tetrolet, and DCNN+Contourlet, and the proposed model are 8.121 dB, 34.687 dB, 34.701 dB, 34.709 dB, 39.479 dB, 39.497 dB, 43.231 dB, 43.480 dB, and 43.491 dB, respectively. The analysis in terms of SSIM is depicted in figure 4.5 c). When the Gaussian noise variance=0.3, the existing techniques, like random, sequential, optimal order, CWSM, SVNN-Wavelet, SVNN-Contourlet, Moth+tetrolet, and DCNN+Contourlet, possesses the SSIM of 0.439, 0.422, 0.429, 0.417, 0.451, 0.4515, 0.456,

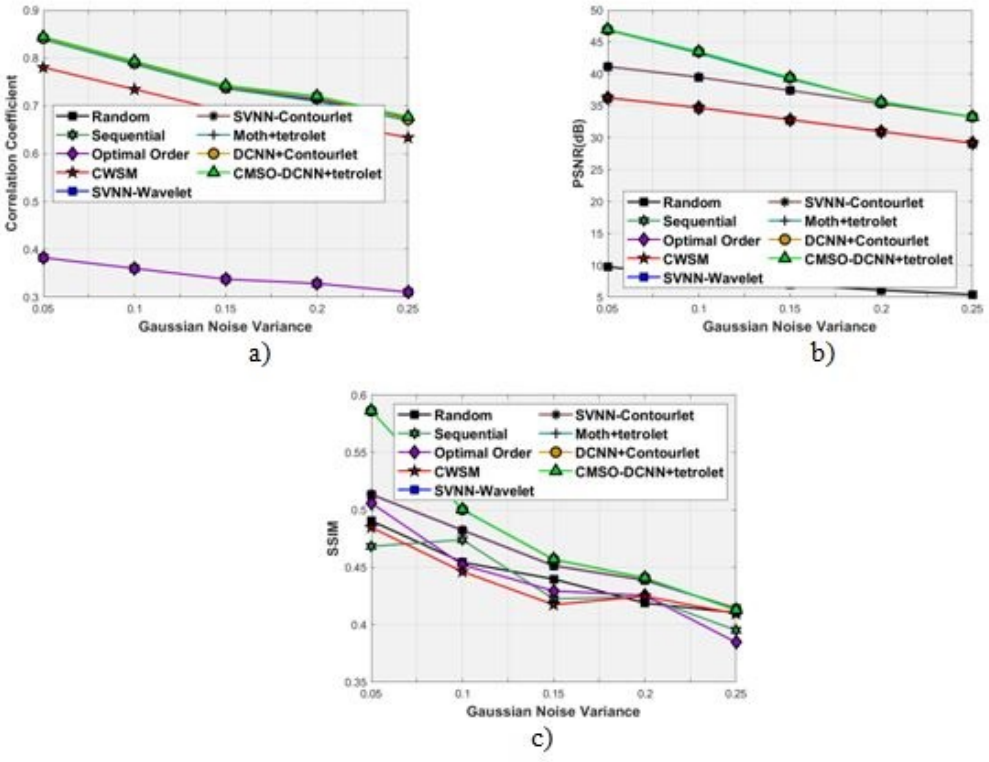

FIG. 4.5. *Comparative analysis by adding Gaussian noise (a) Correlation coefficient, b) PSNR, and (c) SSIM*

and 0.4567, respectively, which is comparatively lower than the CMSO-DCNN+tetrolet. For the same Gaussian noise variance, the developed CMSO-DCNN+tetrolet acquired the SSIM of 0.456.

4.7. Comparative discussion. Table 4.1 depicts the comparative discussion of the existing random, sequential, optimal order, CWSM, SVNN-Wavelet, SVNN-Contourlet, Moth+tetrolet, and DCNN+Contourlet, and the proposed CMSO-DCNN+tetrolet in terms of correlation coefficient, PSNR, and SSIM parameters with impulse, salt and pepper, and Gaussian noise present in the image. The maximum performance measured by proposed CMSO-DCNN+tetrolet in terms of correlation coefficient parameter is 0.85, whereas the correlation coefficient values of existing random, sequential, optimal order, CWSM, SVNN-Wavelet, SVNN-Contourlet, Moth+tetrolet, and DCNN+Contourlet, are 0.691, 0.693,0.708,0.780, 0.842, 0.844,0.822, and 0.846, respectively. The maximal PSNR achieved by the proposed CMSO-DCNN+tetrolet is 46.981dB, whereas the existing random, sequential, optimal order, CWSM, SVNN-Wavelet, SVNN-Contourlet, Moth+tetrolet, and DCNN+Contourlet, acquired the PSNR of 13.748 dB, 39.6 dB, 42.699 dB, 42.704 dB, 42.705 dB, 42.713 dB, 46.86 dB, and 46.935 dB, respectively. The SSIM value computed by proposed CMSO-DCNN+tetrolet is 0.6388, whereas the existing random, sequential, optimal order, CWSM, SVNN-Wavelet, SVNN-Contourlet, Moth+tetrolet, and DCNN+Contourlet, methods acquired the SSIM of 0.490, 0.503, 0.505, 0.491, 0.605, 0.6316, and 0.6319, respectively. It is clearer that the proposed method acquired a maximal correlation coefficient, PSNR, and SSIM.

5. Conclusion. This paper presents the pixel prediction approach based on DCNN classifier and tetrolet transform. This framework employs the medical input image, and the tetrolet Transform is employed for hiding the sensitive information. At first, the best pixels are found out from the image using DCNN classifier, and are trained by CSO and MSO. The Tetrolet transform employs the embedding strength and the coefficient of Transform is employed for embedding the secret message to the input image. At last, the watermark message,

TABLE 4.1
Analysis based on the image with noise

Methods	Correlation coefficient	PSNR (dB)	SSIM
Random	0.691	13.748	0.490
Sequential	0.693	39.6	0.503
Optimal order	0.708	42.699	0.505
CWSM	0.780	42.704	0.491
SVNN-Wavelet	0.842	42.705	0.605
SVNN-Contourlet	0.844	42.713	0.605
Moth+tetrolet	0.822	46.86	0.6316
DCNN+Contourlet	0.846	46.935	0.6319
Proposed CMSO-DCNN+tetrolet	**0.85**	**46.981**	**0.6388**

and the input image gets extracted based on inverse Tetrolet Transform coefficient. The performance of the CMSO-DCNN+tetrolet is evaluated based on correlation coefficient, PSNR, and SSIM. The proposed method produces the maximal correlation coefficient of 0.85, maximal PSNR of 46.981dB, and the maximal SSIM of 0.6388, by applying the impulse, salt and pepper noise, and Gaussian noise in the image that indicates the superiority of proposed method. The future dimension of the research will be concentrated on extending the analysis using other standard databases with highly advanced features.

REFERENCES

[1] P. Eze, U. Parampalli, R. Evans, and D. Liu, *Integrity Verification in Medical Image Retrieval Systems using Spread Spectrum Steganography*, In Proceedings of International Conference on Multimedia Retrieval, (2019), pp. 53-57.

[2] S. Arunkumar, V. Subramaniyaswamy, V. Vijayakumar, N. Chilamkurti, and R. Logesh,*SVD-based Robust Image Steganographic Scheme using RIWT and DCT for Secure Transmission of Medical Images*, Measurement, 139 (2019), pp. 426-437.

[3] R. F. Mansour, Elsaid M. Abdelrahim, *An evolutionary computing enriched RS attack resilient medical image steganography model for telemedicine applications*, Multidimensional Systems and Signal Processing, 30(2) (2017). pp. 791-814.

[4] X. Liao, J. Yin, S. Guo, X. Li, A. K. Sangaiah, *Medical JPEG image steganography based on preserving inter-block dependencies*,Computers and Electrical Engineering, (2017), pp. 1–10.

[5] D. K. Sarmah, and A. J. Kulkarni, *Image Steganography Capacity Improvement Using Cohort Intelligence and Modified Multi-Random Start Local Search Methods*, Arabian Journal for Science and Engineering, 43(8) (2018, pp. 3927-3950.

[6] S. I. Nipanikar, and V. H. Deepthi, *A Multiple Criteria-Based Cost Function Using Wavelet and Edge Transformation for Medical Image Steganography*, Journal of Intelligent Systems, 27(3) (2018), pp. 331-347.

[7] T. Denemark, and J. Fridrich, *Steganography with Multiple JPEG Images of the Same Scene*, IEEE Transactions on Information Forensics and Security, 12(10) (2017), pp. 2308-2319.

[8] D. Hu, L. Wang, W. Jiang, S. Zheng, B. Li, *A Novel Image Steganography Method via Deep Convolutional Generative Adversarial Networks*, IEEE Access, 6 (2018), pp. 38303-38314.

[9] N. G. Kini, Gautam and V. G. Kini, *A Parallel Algorithm to Hide an Image in an Image for Secured Steganography*, In Integrated Intelligent Computing, Communication and Security, (2019), pp. 585-594.

[10] A. Gutub, and M. Al-Ghamdi, *Image Based Steganography to Facilitate Improving Counting-Based Secret Sharing*, 3D Research, 10(1) (2019), pp. 6.

[11] M. M. Hashim, M. S. M. Rahim, F. A. Johi, M. S. Taha, and H. S. Hamad, *Performance evaluation measurement of image steganography techniques with analysis of LSB based on variation image formats*, International Journal of Engineering and Technology, 7(4) (2018), pp. 3505-3514.

[12] K. Muhammad, M. Sajjad, I. Mehmood, S. Rho, and S. W. Baik, *Image steganography using uncorrelated color space and its application for security of visual contents in online social networks*, Future Generation Computer Systems, 86 (2018), pp. 951-960.

[13] B.in Li, M. Wang, X. Li, S. Tan, and J. Huang, *A Strategy of Clustering Modification Directions in Spatial Image Steganography*, IEEE Transactions on Information Forensics and Security, 10(9) (2015), pp. 1905-1917.

[14] G. Linjie, N. Jiangqun, and S. Y. Qing, Uniform Embedding for Efficient JPEG Steganography, IEEE Transactions on Information Forensics and Security, 9 (2014), pp. 814-825.

[15] W. Mazurczyk and L. Caviglione, *Steganography in Modern Smartphones and Mitigation Techniques*, Communications Surveys & Tutorials, IEEE, 17 (2014), pp. 334-357.

[16] Z. Liu, F. Zhang, J. Wang, H. Wang, and J. Huang, Authentication and recovery algorithm for speech signal based on digital watermarking, Signal Processing, 123 (2015), pp. 157-166.

[17] J. Li, X. Li, B. Yang, and X. Sun, *Segmentation-based image copy-move forgery detection scheme*, IEEE Transactions on Information Forensics and Security, 10 (2015), pp. 507-518.

[18] A. Khan, A. Siddiqa, S. Munib, and S. A. Malik, *A recent survey of reversible watermarking techniques*, Information

Sciences, 279 (2014), pp. 251-272.
[19] S. M. M. Karim, S. Rahman, I. Hossain, *New Approach for LSB Based Image Steganography using Secret Key*, In Proceedings of 14th International Conference on Computer and Information Technology, (2011), pp. 22-24.
[20] Y. Yeung, W. Lu, Y. Xu, J. Chen, and R. Li, *Secure binary image steganography based on LTP distortion minimization*, Multimedia Tools and Applications, 2019, pp. 1-22.
[21] W. Tang, B. Li, S. Tan, M. Barni, and J. Huang, *CNN-based Adversarial Embedding for Image Steganography*, IEEE Transactions on Information Forensics and Security, 14(8) 2019, pp. 2074-2087.
[22] S. Chakraborty, A. S. Jalal, and C. Bhatnagar, *LSB based non blind predictive edge adaptive image steganography*, Multimedia Tools and Applications, 76(6) 2017, pp. 7973-7987.
[23] A. Miri, and K. Faez, *An image steganography method based on integer wavelet transform*, Multimedia Tools and Applications, 77(11) 2018, pp. 13133-13144.
[24] G. S. Lin, Y. T. Chang, and W. N Lie, *A framework of enhancing image steganography with picture quality optimization and anti-steganalysis based on simulated annealing algorithm*, IEEE Transactions on Multimedia, 12(5) (2010), pp. 345-357.
[25] J. Krommweh, *Tetrolet Transform: A New Adaptive Haar Wavelet Algorithm for Sparse Image Representation*, Journal of Visual Communication and Image Representation, 21(4) 2010, pp. 364-374.
[26] Pandey, Prateekshit, R. Singh, and M. Vatsa, *Face recognition using scattering wavelet under Illicit Drug Abuse variations*, In Proceedings of International Conference on Biometrics (ICB), Halmstad, Sweden, (2016), pp. 1-6.
[27] W. K. Kong, D. Zhang, and W. Li, *Palmprint feature extraction using 2-D Gabor filters*, Pattern recognition, 36(10) (2003), pp. 2339-2347.
[28] Z. Guo, L. Zhang and D. Zhang, *A Completed Modeling of Local Binary Pattern Operator for Texture Classification*, IEEE Transactions on Image Processing, 19(6) (2010), pp. 1657-1663.
[29] F. Tu, S. Yin, P. Ouyang, S. Tang, L. Liu, and S. Wei, *Deep Convolutional Neural Network Architecture With Reconfigurable Computation Patterns*, IEEE Transactions on very large scale integration (VLSI) systems, 25(8) 2017, pp. 2220 - 2233.
[30] BRATS brain tumor database, *https://www.smir.ch/BRATS/Start2015*, Accessed on (2019).
[31] S. Islam, M. R. Modi and P. Gupta, *Edge-based image steganography*, EURASIP Journal on Information Security, 2014(1) (2014).
[32] M. Ramalingam and N. A. M. Isa, *A steganography approach for sequential data encoding and decoding in video images*, In proceedings of International Conference on Computer, Control, Informatics and Its Applications (IC3INA), Bandung, Indonesia, (2014), pp. 120-125.
[33] B. Li, M. Wang, X. Li, S. Tan and J. Huang, *A Strategy of Clustering Modification Directions in Spatial Image Steganography*, IEEE Transactions on Information Forensics and Security, 10(9) (2015), pp. 1905-1917.
[34] S. Nikolov, P. Hill, D. Bull, and N. Canagarajah, *Wavelets for image fusion, In: Petrosian A.A., Meyer F.G. (eds) Wavelets in Signal and Image Analysis*, Computational Imaging and Vision, 19 (2001), pp. 213-241.
[35] S. Nipanikar, V. H. Deepthi, *A Multiple Criteria-Based Cost Function Using Wavelet and Edge Transformation for Medical Image Steganography*, Journal of Intelligent Systems, 27(3) (2016).
[36] G. -G. Wang, *Moth search algorithm: a bio-inspired metaheuristic algorithm for global optimization problems*, Memetic Computing, 10(2) (2018), pp. 151-164.
[37] X. Meng, Y. Liu, X. Gao, and H. Zhang, *A New Bio-inspired Algorithm: Chicken Swarm Optimization Xianbing*, In proceedings of International conference in swarm intelligence, (2018), pp. 86-94.
[38] V. Vinolin and S. Vinusha, *Edge-based Image Steganography using Edge Least Significant Bit (ELSB) Technique*, Multimedia Research, 1(1) (2018), pp. 9-16.
[39] S. Vinusha, *Secret Image Sharing and Steganography Using Haar Wavelet Transform*, Multimedia Research, (2019), 2(2), pp. 28-34.
[40] M. Mukhedkar, P. Powar, and P. Gaikwad, *Secure non real time image encryption algorithm development using cryptography & Steganography*, In Proceedings of the Annual IEEE India Conference (INDICON), (2015), pp. 1-6.

Edited by: P. Vijaya
Received: Dec 9, 2019
Accepted: Jun 22, 2020