Fuzzy proximity-based robust data hiding scheme with interval threshold

Prabhash Kumar Singh1,2 · Biswapati Jana1 · Kakali Datta2

Accepted: 29 July 2022 / Published online: 11 October 2022
© The Author(s), under exclusive licence to Springer-Verlag GmbH Germany, part of Springer Nature 2022

Abstract
Secret communication of sensitive data must progress in a trustworthy environment through data hiding. Using Mamdani fuzzy logic to identify color proximity at the block level and a shared secret key and post-processing system, this paper attempts to develop a robust data hiding scheme with similarity measures to ensure good visual quality, robustness, imperceptibility and enhance the security. In accordance with the Gestalt principle, proximity among the nearby objects is higher, whose value varies from expert to expert. Therefore, a possibility for type-I fuzzy logic to be used to evaluate proximity. Fuzzy proximity is computed by means of a difference in intensity (colordiff) and distance (closeness). Further, the block color proximity obtained from the proximity calculation network is graded using an interval threshold. Accordingly, data embedding is processed in the sequence generated by the shared secret keys. The tampering coincidence problem is solved through a post-processing approach to increase the quality and accuracy of the recovered secret message. The experimental analysis, steganalysis and comparisons clearly illustrate the effectiveness of the proposed scheme in terms of visual quality, structural similarity, recoverability and robustness.

Keywords Data hiding · Fuzzy logic · Proximity · Tampering · Steganalysis

1 Introduction
The COVID-19 pandemic has made us realize the importance of the Internet and multimedia technologies. The complete communication system worldwide has moved to the Internet using multimedia data. They have been the actual source of data transmission and information exchange (Wong et al. 2021). It is imperative to say that the trend is far from over. Thus, a reason to think more about the security, authenticity, robustness and illegal copying of the documents exchanged through open online media. Popular image processing software can modify digital multimedia documents quickly and professionally. This has aided in the creation of various image forgery attacks. Authentication has been necessary for tamper detection and owner identification in many human-centric applications, including forensic analysis, insurance processing, surveillance systems, radiography imaging, journalism, etc. Data hiding plays a critical role in thwarting unwanted activities carried out with advanced technologies.

Technically, hidden data communication can be achieved through cryptography, watermarking and steganography with a motive to provide information security. Confidentiality and security of the information are the souls of data communication. Complete data hiding systems can be put into two categories: information hiding and encryption. Generally, watermarking is applied to hide or embed a message in a cover image so that authenticity and copyright can be proved when asked for (Fkirin et al. 2021). The watermark applied can be visible or invisible. On the other hand, steganography has the power to conceal a hidden secret message in a covered medium, making it impossible for an attacker to discover its presence. The hidden message is used for various purposes, including secret data sharing, secure correspondence and authentication. Cryptography does not hide the
secret data but encrypts the original data through a secret key and then sends it to the receiver.

The objective of any data hiding scheme is better to feature robustness, imperceptibility, security and hiding capacity. However, it is arduous to extract the maximum out of each mentioned feature at any point in time. There have to be pareto-optimal solutions to maintain each property at its optimum level. An increase in the amount would lead to a decrease in the other. Also, at times, the development of a data hiding scheme is lead considering the field of application. In some fields, robustness is preferred, while some prioritize high security and imperceptibility. As suggested by Abraham et al. (2004), robustness is generally required for a system where the chance of unwanted malicious attacks is high. High visual quality or capacity is not always required since a watermark can be either visible or invisible within watermarking.

In this paper, an emphasis has been put on designing a robust data hiding scheme that follows the Gestalt principle governing color proximity. It states “things that are close together appear to be more related than things that are spaced farther apart”. The relationship between items cannot be predicted in a precise way. Thus, a fuzzy model has been designed to counter the imprecise nature of color proximity related to a group of pixels. The degree of proximity may vary from expert to expert. Therefore, to model an ambiguous nature of proximity in a real-world scenario, fuzzy sets (Zadeh 1965) and logic have been applied rather than conventional logic. The proximity between the pixels is measured through fuzzy logic by simulating not only the difference between the intensity of the pixels but also the distance apart they lie. Additionally, the scheme is also formulated to address the problem of tampering coincidence (Lee and Lin 2008) which causes a great hindrance in the path of recovery of the secret data after tampering. When both the blocks containing original and recovery data are tampered, it becomes difficult to extract the hidden information. To counter this problem, authors have embedded copies of the secret message at random blocks. As a result, even after tampering with a specific region, the message could be possibly obtained from some other blocks. Consequently, a second chance to reconstruct the hidden secret data or image should be applied in case of deleting a specific region and recovering secret data. The main contributions of the proposed scheme are as follows:

i) A novel steganographic procedure has been built with fuzzy logic to hide secret data.

ii) Fuzzy rule-based controller is designed that can efficiently compute proximity relationship between pixels based on two properties, color difference and closeness.

iii) Two copies of secret data are hidden separately at random positions according to the secret key to avoid tampering coincidence problem.

iv) Color difference and closeness between a pair of pixels is modeled through fuzzy linguistic variables.

v) Data hiding scheme formulated is semi-fragile and recoverable with high visual quality measured by PSNR (dB) and SSIM than recent state-of-the-art schemes.

In this paper, few key terms that will come across more often are explained here and a list of notations used are illustrated in Table 1 for easy readability.

| Notations | Descriptions |
|-----------|-------------|
| Δ_RGB | Difference in intensity (colordiff) |
| ψ | Euclidean distance (closeness) |
| α | Bias weight |
| R_i | ith fuzzy rule |
| χ | Proximity of pixel |
| χ_cpr | Cumulative proximity of pixel |
| ξ | Block proximity |
| [t−, t+] | Interval threshold |
| [t− + ϵ, t+ + ϵ] | Interval Threshold Range |
| S | Strong graded blocks |
| M | Moderate graded blocks |
| W | Weak graded blocks |
| Key_S | Secret key for S blocks |
| Key_M | Secret key for M blocks |
| Key_W | Secret key for W blocks |
| O_1, O_2 | Vector to store extracted secret bits |
| CI | Cover image |
| SI | Secret image |
| STI | Stego image |
| ESI | Extracted secret image |
| RSI | Recovered secret image |

The rest of the paper is organized as follows: Sect. 2 carries a data hiding related work relevant to this paper. In Sect. 3, the preliminary concept of fuzzy logic is discussed. The proposed embedding and extraction procedures are provided in
Sect. 4, followed by experimental results and comparisons in Sect. 5. In the end, a conclusion is drawn in Sect. 6.

2 Related work

The relevant investigation regarding hidden data communication has progressed over time. In the literature, many data-hiding-based authentication methods have been reported by many researchers recently. Three main characteristics distinguish authentication methods: (1) data generation for authentication, (2) data generation for recovery and (3) the data embedding process used. In multimedia authentication, fragile data hiding schemes incorporating Least Significant Bit (LSB) replacement are commonly used. For any data hiding scheme, imperceptibility is a major concern. The message must be embedded in such a way that its presence is not tracked either statistically or visually (Yang et al. 2022).

In steganography, the information required for communication to the receiver is embedded in the cover image by a sender. On receiving, a receiver segregates the confidential data. Two types of motivations can be used to design a steganographic process: reversible and irreversible. In the former, both the cover and message can be recovered precisely, while in the latter, only the message is recovered without error.

The most widely used reversible schemes are based on histogram shifting and error expansion (Jia et al. 2019; Lee et al. 2019). Most reversible methods are unaware of the importance of security and robustness considerations in certain situations. The least significant bit (LSB) technique is one of the most straightforward and widely used spatial image steganographic methods. The hidden data are inserted directly into the host image using LSB-based spatial domain techniques, which modify the least significant bits of selected pixels without distorting the visual quality of the original cover image. Earlier LSB steganography research (Chandramouli and Memon 2001; Sutaone and Khandare 2008) focused solely on designing the device to maximize payload capability by using the majority of the cover image pixels. The research issue became more oriented after that to develop sophisticated, robust LSB-based cryptography-steganography that can withstand such steganalysis attacks (Patel and Meena 2016; Rajendran and Doraipandian 2017; Shafi et al. 2018). Learning methods, for example, were used to optimize LSB substitution (Maity and Kundu 2009; Dadgostar and Afsari 2016).

Hidden bits can also be integrated into a cover image using transform domain values. The secret bits are concealed under the sub-band frequency coefficients in transform domain methods (Valandar et al. 2020). In the field of steganography, a variety of transform domain methods are used, with the most common schemes being the Discrete Fourier Transform (DFT), Discrete Cosine Transform (DCT), Discrete Wavelet Transform (DWT) and variations in these basic transforms. In the field of DFT steganography, various image systems have been proposed (Khashandarag et al. 2011; Haibo 2008). The paper (Seki et al. 2005) shows a steganographic image scheme with hidden data embedded in the jpeg encoder. In steganography based on DCT, the cover image is divided into non-overlapping (8 × 8) pixel blocks. A similar modified DWT scheme for embedding hidden bits in between wavelet coefficients was discussed by Samer et al. (2017). Compared to the traditional DWT approach, Diamond Encoding (DE) in DWT is proposed here to improve protection and reduce image distortion.

The true motive for steganography is to share data in an undetectable way. Researchers experimented with various ways to improve imperceptibility in the steganographic system. Chang et al. (2008) used a 2-level quantization method along with a genetic algorithm in which three bits are additionally embedded over image blocks with high fidelity to try to maximize payload power. For the embedding operation, this method uses the most suitable regions or features in the cover image (Balasubramanian et al. 2014; Hamid et al. 2012a, b). Local features are derived based on image segmentation that can be used to differentiate noisy areas, with the embedding focused in the noise areas (Niimi et al. 1999). A threshold value is used to pick the high frequency pixels of the cover and then the LSBM Revisited algorithm is used to conceal hidden data (Mungmode et al. 2016). However, finding the optimal local solution requires many computational stages, which slows down the search for specific embedding locations. As a result, the genetic algorithm is highly resistant to image manipulation.

A fuzzy logic-based method focuses more on visual consistency preservation to enhance stego-media imperceptibility at the expense of increased modeling complexity. Fuzzy techniques have been used in a variety of ways, for example, a Fuzzy Inference System (FIS) with HVS is used in a study (Chang et al. 2008) to make decisions based on local statistical, texture and brightness information-based feature vectors. The method will specify the semantic rules for the embedding process using these features from cover image sub-regions. Also, at a higher embedding rate, the principle helps to minimize stego image distortions. Few recent investigator (Jagadeesh et al. 2016; Tang et al. 2021, 2022) illustrates a different fuzzy-based approach. Before the actual embedding operation, the cover pixel selection is based on fuzzy pixel classification and the secret message is converted to a mode of fuzzy data. Kiani et al. (2009) clarify another fuzzy-based watermarking technique that employs a fuzzy-c clustering algorithm based on transform domain derivative features in various directions. Fuzzy logic can improve steganographic schemes in various ways, particularly when the image textures are vague and/or ambiguous. It benefits the system by
quickly identifying appropriate image patterns and reducing irreversible complexities. This will ultimately aid practical applications that make use of sufficient imperceptibility. Some of the earlier work relies on a correlation-based cover selection method, image block similarity (Sajedi and Jamzad 2008) and statistical measures features.

2.1 Review of Ashraf et al. (2020) Scheme

Based on human perception, Ashraf et al. formulated a rule-based interval type-2 fuzzy logic system to calculate a similarity measure of the surrounding pixels. LSB approach was considered to find the pixel values with high similarity measures. A 3 × 3 window of pixels corresponding to the central pixel was taken into account for similarity calculation. The complete procedure involved five steps: calculation of gray level pixel difference, conversion of the value to type-2 fuzzy, setting up the inference engine, defuzzification and finally, similarity calculation. The authors employed a similarity threshold to choose the best pixel for embedding. At the time of embedding, the actual pixel value was converted with an image quality of 5125 bpp on average for the standard images. It has the scope of enhancing the visual quality of the image. This scheme has neither been tested for the color images nor any geometrical or non-geometrical attacks.

From the overview of the discussion mentioned above, fuzzy logic has been put to its strength in this paper to design a robust data hiding scheme to model the vague nature associated with the features of an image.

3 Preliminaries

The universal set can be described so that all elements are categorized as members or nonmembers based on a predefined characteristic feature. If U denotes the universe set and x denotes the general elements, the characteristic function F_X(x) maps all members of U into the set [0, 1]. The classical sets can be represented mathematically as membership functions by the following expression:

\[ F_X(x) = \begin{cases} 1 & \text{if } x \text{ belongs to } X \\ 0 & \text{if } x \text{ does not belong to } X. \end{cases} \]  

(1)

Fuzzy set: It is a set on which a mathematical model can be constructed to represent imprecise or vague situation. It derives a degree of association of domain elements to its fuzzy set. Fuzzy sets considers all possible values that can exist between 0 and 1 that is yes or no. It does not directly make a distinction between those that are part of the collection and those that are not. Fuzzy set tags a membership number which defines the extent of association of an object in a set. Let x ∈ X be an element present in a fuzzy set A. The degree of membership of x in A is indicated by \( \mu_A(x) \). Subsequently, the fuzzy set is represented in terms of ordered pair \( A = (x, \mu_A(x)/x \in X) \).

Fuzzy number: A number which is imprecise in nature such that it belongs to real number \( \Re \) and have the weight, i.e., membership value between the range [0, 1]. It has a convex and continuous nature.

For example, Let \( A_1 \) and \( A_2 \) be two fuzzy numbers in \( \Re \) with membership functions \( \mu_{A_1}(x) \) and \( \mu_{A_2}(x) \), respectively. Then, according to Dubois (1980) and Zadeh (1978)

\[ \text{pos}(A_1 \ast A_2) = \sup\{\min(\mu_{A_1}(x), \mu_{A_2}(y)), x, y \in \Re, x \ast y\} \]  

(2)

where pos represents possibility, \( \ast \) is any one of the relations >, <, =, ≤ and \( \Re \) represents set of real numbers.

A basic unit of a fuzzy number that conforms to the above description can be depicted graphically in Fig. 1. The numbers \( a_1, a_2, a_3 \) and \( a_4 \) plotted in the given figure are real numbers whose membership values are calculated. The membership function determines whether the fuzzy number \( \hat{A} \) is continuous or discrete. A continuous fuzzy number can be represented by a special class of triangular fuzzy numbers (Roy and Bhaumik 2018), trapezoidal fuzzy numbers and Gaussian fuzzy numbers.

Triangular fuzzy number (TFN): A TFN \( \hat{A} = (a_1, a_2, a_3) \) (refer Fig. 2) has three parameters \( a_1, a_2, a_3 \), where \( a_1 \leq a_2 < a_3 \) and is characterized by the membership function \( \mu_{\hat{A}} \) given by Eq. (3).

\[ \mu_{\hat{A}}(x) = \begin{cases} \frac{x-a_1}{a_2-a_1} & \text{for } a_1 \leq x \leq a_2 \\ \frac{a_3-x}{a_3-a_2} & \text{for } a_2 \leq x \leq a_3 \\ 0 & \text{otherwise} \end{cases} \]  

(3)
Fuzzy proximity-based robust data hiding scheme... 4015

Linguistic variables: The concept of linguistic variables is crucial to the growth of fuzzy set theory (Bhaumik et al. 2020). Fuzzy logic is mainly used to measure and reason out imprecise or ambiguous words found in our languages. Linguistic or fuzzy variables are the terminology used to describe these terms. For example, temperature could be denoted as a linguistic variable such as very hot, hot, cold, and very cold.

Fuzzy rule-based system: Expert information is incorporated into the Mamdani fuzzy rule-based structure in the form of linguistic variables in this paper. The fuzzy rules are made up of input and output variables that take values from term sets that represent the meanings of each linguistic term. Every rule is a condition-action statement with a human-readable interpretation. It has the structure of disjunctive normal form fuzzy rule, which has the following form:

\[
\text{If } X_1 \text{ is } A_1 \text{ and } \ldots \text{ and } X_n \text{ is } A_n \text{ then } Y \text{ is } B. \quad (4)
\]

4 Proposed scheme

The proposed data hiding scheme can be confined to two stages. The first stage consists of block segregation, fuzzy proximity calculation, block proximity gradation with interval threshold and data embedding. Figure 3 depicts the flowchart of the stages considered in the designed embedding technique. During the second stage, secret data extraction, authentication, post-processing and self-recovery are performed.

4.1 Block segregation

In the designed scheme, a color image of size \((M \times N)\) is taken into account and split into three separate channels of red, green and blue. Each channel is further partitioned into non-overlapping blocks of size \((b \times b)\). The blocks are used to measure the proximity relationship among the pixels present within the blocks. In the literature, authors have used similarity to measure the relationship between the pixels. However, Wuerger et al. (1995) have iterated in their research that proximity is not dependent only on the Euclidean properties of the pixels in contention. The proximity between the pixels follows the law of proximity as defined in the Gestalt grouping law. This simple rule states that objects close together are more likely to be grouped than those further apart objects. Thus, in the proposed research, while calculating the proximity of pixel \((\chi)\), importance has been given to the difference in intensity (colordiff) and distance (closeness) as well.

\[
\chi(P_i, P_j) = \alpha \Delta_{\text{RGB}}(P_i, P_j) + (1 - \alpha)\psi(P_i, P_j) \quad (5)
\]

\[
\Delta_{\text{RGB}}(P_i, P_j) = \sqrt{(R_i - R_j)^2 + (G_i - G_j)^2 + (B_i - B_j)^2} \quad (6)
\]

\[
\psi(P_i, P_j) = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2} \quad (7)
\]

Consider a pixel \(P_i\) and \(P_j\) with location \((x_i, y_i)\) and \((x_j, y_j)\), respectively, in a block as shown in Fig. 4. The proximity of pixel \(P_i\) in reference to pixel \(P_j\) is given by Eq. (5). Here, \(\Delta_{\text{RGB}}(P_i, P_j)\) represents the difference in intensity values of red, green and blue channels as provided in Eq. (6). The closeness of two pixels is measured in terms of Euclidean distance between them through Eq. (7). The parameter \(\alpha\) is the bias weight added to control the relative importance among the color difference and closeness.

4.2 Fuzzy proximity calculation

To calculate proximity, a fuzzy rule-based controller is prepared with four functions namely (i) Fuzzification, (ii) Fuzzy rule base, (iii) Fuzzy inference mechanism and iv) Defuzzification. The perceptual analysis of proximity may vary from expert to expert, as has been shown by Demirci (2006) in its paper for similarity calculation. Thus, it is best to use fuzzy rule-based logic to deal with the imprecise nature of proximity between the pixels. To calculate proximity between the pixels, two factors, color differences (colordiff) and closeness, are considered input for fuzzification. Each factor is represented as a fuzzy set with fuzzy linguistic terms as low, medium and high. There are several membership functions to model these linguistic terms, as trapezoidal, Gaussian, triangular, etc., however, the triangular fuzzy number is chosen to represent the stated linguistic terms due to its simplicity and symmetric representation. Also, during the experimental analysis, the triangular membership function maintained high visual quality with good PSNR than the other stated membership functions. According to Eq. (6), the minimum and maximum values that it can hold are 0 and 255\(\sqrt{3}\), respectively, since the difference in a color channel between two pixels is in the range \([0, 255]\). In the same way, closeness has the value in the range \([1, \sqrt{2}(b - 1)]\). Both these factors are modeled as shown in Fig 5.
After fuzzification, a fuzzy inference engine is set up which performs according to certain rules. The fuzzy inputs generated after fuzzification are combined with rule base to produce a degree of association between two pixels. This association measures how strong a proximity exists. So, given $n$ inputs such that $x_1 \in X_1$, $x_2 \in X_2$, ..., $x_n \in X_n$ and one output $y \in Y$, the $i^{th}$ fuzzy rule according to Eq. (4) is represented as

$$R_i: \text{If } x_1 \text{ is } \tilde{A}_1, \text{ x}_2 \text{ is } \tilde{A}_2, \ldots, \text{ x}_n \text{ is } \tilde{A}_n \text{ Then } y \text{ is } \tilde{B}$$

The set of inputs in the rule base is also called antecedent and output as consequent. The best fuzzy rules that performed well for the proposed work are illustrated in Table 2. The consequent is generated in the form of $\chi$, which is denoted by the linguistic term Low, Medium and High. The triangular fuzzy number of the consequent is shown in Fig. 6 with overlapping in the interval $[0 - 100]$ with 0 being the minimum and 100 as the maximum.

The actions of an individual rule are superimposed to construct a fuzzy output set as the final production. The
Table 2  Fuzzy rules

| Rule | Antecedent | Consequent |
|------|------------|------------|
| \( R_1 \) | If color difference is low and closeness is high | then \( \chi \) is high |
| \( R_2 \) | If color difference is low and closeness is medium | then \( \chi \) is medium |
| \( R_3 \) | If color difference is low and closeness is low | then \( \chi \) is low |
| \( R_4 \) | If color difference is medium and closeness is high | then \( \chi \) is high |
| \( R_5 \) | If color difference is medium and closeness is medium | then \( \chi \) is medium |
| \( R_6 \) | If color difference is medium and closeness is low | then \( \chi \) is low |
| \( R_7 \) | If color difference is high and closeness is high | then \( \chi \) is high |
| \( R_8 \) | If color difference is high and closeness is medium | then \( \chi \) is medium |
| \( R_9 \) | If color difference is high and closeness is low | then \( \chi \) is low |

![Fig. 6 Membership functions for proximity of pixels (\( \chi \))](image)

![Fig. 7 Proximity calculation network](image)

Fuzzy output is generated by combining the outputs of the rules that have been shot. On the fuzzy output, the following defuzzification methods generate a crisp value. The crisp value is the point of the output membership function that divides the region in half, as the name implies. Maxima methods and area-based methods are two popular defuzzification techniques. The centroid method is a common area-based defuzzification technique.

In this way, for each \( P_i \) present in a block, we calculate the proximity pixels \( \chi(P_i, P_j) \) such that \( j = 1, 2, \ldots, b^2; j \neq i \). The proximity calculation network of pixels in a block is demonstrated in Fig. 7. At last, the cumulative proximity of pixel \( \chi_{cpr}(P_i) \) is computed by Eq. (8).

\[
\chi_{cpr}(P_i) = \frac{\sum_{j=1}^{b^2} pr(P_i, P_j)}{b^2 - 1}; j \neq i
\]  

(8)

Now, the proximity of the block (\( \xi \)) in whole is determined by Eq. (9) as follows:

\[
\xi = \frac{1}{b^2} \sum_{i=1}^{b^2} \chi_{cpr}(P_i)
\]  

(9)

Accordingly, the proximity of each block needs to be determined.

4.3 Block proximity gradation with interval threshold

After calculation of block proximity (\( \xi \)), the blocks are graded as strong (\( S \)), moderate (\( M \)) and weak (\( W \)) based on proximity value, \( \xi \), using interval threshold. In the literature, a threshold value is generally precise, which has the advantage of partitioning an item into either of two sets. However, in this paper, an interval threshold has been selected for the study. An interval threshold contains an interval number between assigned ranges to partition a given item into any of three explicitly defined sections. Let \( t = [t^-, t^+] \) be an interval threshold in the range \( [0, 1] \) such that \( 0 < t^-, t^+ < 1 \). A parameter \( \epsilon \) is introduced to extend a degree of elasticity within the interval threshold so that threshold values \( t^- \) and \( t^+ \) do not behave rigidly. The value of \( \epsilon \) may vary from image to image. Now, the interval threshold becomes \( t = [t^- + \epsilon, t^+ + \epsilon] \). Before categorization, the value of \( \xi \) is normalized between 0 and 1. A block \((i)\) is graded as given in Eq. (10) and represented in Fig. 8.

\[
\text{block}(i) = \begin{cases} 
S & \text{if } \xi \leq (t^- + \epsilon) \\
W & \text{if } \xi \geq (t^+ + \epsilon) \\
M & \text{otherwise}
\end{cases}
\]  

(10)
4.4 Data embedding

For data embedding, similar graded blocks are grouped to hide data in the LSB. Blocks categorized as strong, moderate and weak are embedded with 1, 2 and 3 bits, respectively. Moreover, a set of three secret keys, Key_S, Key_M and Key_W, are generated to determine the sequence of blocks for data hiding in the corresponding red, green and blue channel. A secret image considered for embedding within the cover image is converted into a binary bitstream. Each bit of the secret image is embedded twice in the appropriate channels too. As given in Fig. 9(iii), the first secret bit 1 is embedded in the red channel of 4th block and the green channel of 10th block. The second bit, 0, is embedded in the blue and red channel of 15th and 6th blocks, respectively. Further, 2-bit embedding (0, 1) is carried out in two channels, such as the red and green channels of blocks 14th and 11th. Similarly, 3-bit embedding (0, 1, 0) is done in all three channels of a block, but in sequence, as appears, during data hiding such as blue, red and green channel of block 16th and 8th. Thus, in this way, total data embedding of secret data is completed and, finally, the stego image is generated. A pseudocode of the given technique is provided in Algorithm 1.

4.5 Data extraction

At the time of extraction, the stego image is passed through a fuzzy model and a block proximity matrix is generated, similar to data embedding. Data extraction from the LSB bit of channels is done in the same sequence and blocks as generated by the shared secret keys. Since a bit was double embedded during embedding, two vectors O_1 and O_2 are taken to maintain the copies of embedded bits. A extracted secret image is formed using either O_1 or O_2.

Further, a comparison is made between O_1 and O_2 to find any dissimilarity. For a dissimilarity, post-processing is applied as follows: A position i where a mismatch is found between O_1 and O_2, three positions prior and three positions after i is checked to predict the pattern of 1 and 0. Accordingly, the recovered secret image is composed of enhanced features.

5 Experimental results and comparison

The performance and efficiency of the proposed scheme are tested on a set of standard images collected across four databases, namely the UCID database (UCID 2017), USC-SIPI database (USC-SIPI 2017), Kodak database and STARE...
Fuzzy proximity-based robust data hiding scheme...

| Block Proximity Matrix |
|------------------------|
| S | 1 | M | 2 | S | 3 | S | 4 |
| W | 5 | S | 6 | M | 7 | W | 8 |
| W | 9 | S | 10 | M | 11 | W | 12 |
| S | 13 | M | 14 | S | 15 | W | 16 |

**Embedding Sequence**

- $S = \{1, 3, 4, 6, 10, 13, 15\}$
- $M = \{2, 7, 11, 14\}$
- $W = \{5, 8, 9, 12, 16\}$

**After Shuffle using keys**

- $S'(\text{Key}_S) = \{4, 15, 13, 10, 6, 1, 3\}$
- $M'(\text{Key}_M) = \{14, 7, 11, 2\}$
- $W'(\text{Key}_W) = \{16, 12, 8, 9, 5\}$

**Fig. 9** Example of Data Embedding

**Fig. 10** Set of considered cover images from standard databases for experiment

| UCD IMAGE DATABASE |
|--------------------|
| Summer | Christmas | Camel | Street |

| URC. 5N IMAGE DATABASE |
|-------------------------|
| Airplane | Car | Tree | Flowers |

| KRPAK IMAGE DATABASE |
|----------------------|
| Sunset | Beach | Snow | Forest |

| STARE IMAGE DATABASE |
|----------------------|
| Sunrise | Sunset | Clouds | City |

The results are computed for structural similarity index measurement (SSIM), peak signal-to-noise ratio (PSNR) and universal quality index (Q-Index). These three parameters help to analyze the performance of stego images. It shows the amount of perceptual effect caused due to a change in the number of bits of the original cover image. Other parameters such as standard deviation (SD), correlation coefficient (CC), normalize correlation coefficient (NCC) and bit error rate (BER) are computed to measure the distortion in stego images.

### 5.1 Quality of stego image and payload

After data hiding of the secret bits, as mentioned in the previous section, the quality of the stego image is checked through the parameters PSNR and Q-Index. PSNR is a measurement tool that is shown to be composed of the error squared value as the key component based on Eqs. (11) and (12). Here, $M$ and $N$ represent the resolution and $C$ depicts the number of channels of the image. $I_c(i, j, k)$ and $I_s(i, j, k)$ are the intensity value of cover and stego image, respectively, at coordinates $(i, j)$ and channel $k$. The difference in pixel values at the same coordinates and channels generates the error value. If there are more variations in the pixel values between the two images, PSNR will produce a smaller value, but if there are more differences in the pixel values, PSNR will produce a higher value. Q-Index, the universal image quality index, is used to judge distortion caused relative to three combination factors: loss of luminance distortion, contrast distortion and correlation.

\[
\text{MSE} = \frac{1}{M \times N \times C} \sum_{i=1}^{M} \sum_{j=1}^{N} \sum_{k=1}^{C} (I_c(i, j, k) - I_s(i, j, k))^2
\]  

\[
\text{PSNR} = 10 \log_{10} \left( \frac{\text{MAX}^2}{\text{MSE}} \right)
\]  

To measure the quality and payload of the stego image, a precise analysis has been conducted on three different sizes of the cover image: $(64 \times 64)$, $(128 \times 128)$ and $(512 \times 512)$. The results obtained are revealed in Tables 3, 4 and 5. On observation, it can be stated that the PSNR value lies between 64 dB and 68 dB and Q-Index maintains a consistent quality of 0.9999. This acknowledges the high quality of the stego image obtained after data hiding. The average PSNR of the stego image acquired is 66 dB.

Also, Tables 3, 4 and 5 show the capability and payload achieved in each of the three cases described. The number of
Table 3  Capacity, PSNR, Q-index and payload values of (64 × 64) stego images

| Dataset | Image     | Capacity (bits) | PSNR (dB) | Q-Index | Payload (bpp) |
|---------|-----------|----------------|-----------|---------|---------------|
| UCID    | ucid00022 | 441            | 65.21     | 0.9999  | 0.04          |
|         | ucid00075 | 486            | 65.52     | 0.9999  | 0.04          |
|         | ucid00102 | 543            | 64.43     | 0.9999  | 0.04          |
|         | ucid00646 | 613            | 64.04     | 0.9999  | 0.05          |
|         | **Average** | **521**        | **64.80** | **0.9999** | **0.04** |
| USC_SIPI| Aeroplane  | 435            | 65.62     | 0.9999  | 0.04          |
|         | Baboon     | 409            | 66.04     | 0.9999  | 0.03          |
|         | Boat       | 524            | 65.01     | 0.9999  | 0.04          |
|         | Peepers    | 481            | 65.64     | 0.9999  | 0.04          |
|         | **Average** | **452**        | **65.87** | **0.9999** | **0.04** |
| Kodak   | kodim04    | 382            | 65.79     | 0.9999  | 0.03          |
|         | kodim05    | 495            | 65.24     | 0.9999  | 0.04          |
|         | kodim15    | 387            | 66.31     | 0.9999  | 0.03          |
|         | kodim23    | 405            | 65.99     | 0.9999  | 0.03          |
|         | **Average** | **417**        | **65.83** | **0.9999** | **0.03** |
| STARE   | im0005     | 364            | 66.96     | 0.9999  | 0.03          |
|         | im0034     | 402            | 66.06     | 0.9999  | 0.03          |
|         | im0086     | 331            | 66.93     | 0.9999  | 0.03          |
|         | im0280     | 358            | 66.38     | 0.9999  | 0.03          |
|         | **Average** | **364**        | **66.58** | **0.9999** | **0.03** |

Table 4  Capacity, PSNR, Q-Index and Payload values of (128 × 128) stego images

| Dataset | Image     | Capacity (bits) | PSNR (dB) | Q-Index | Payload (bpp) |
|---------|-----------|----------------|-----------|---------|---------------|
| UCID    | ucid00022 | 1942           | 65.37     | 0.9999  | 0.04          |
|         | ucid00075 | 1860           | 65.54     | 0.9999  | 0.04          |
|         | ucid00102 | 1858           | 65.36     | 0.9999  | 0.04          |
|         | ucid00646 | 2006           | 65.09     | 0.9999  | 0.04          |
|         | **Average** | **1917**       | **65.34** | **0.9999** | **0.04** |
| USC_SIPI| Aeroplane  | 1559           | 66.11     | 0.9999  | 0.03          |
|         | Baboon     | 1767           | 65.46     | 0.9999  | 0.04          |
|         | Boat       | 1680           | 65.86     | 0.9999  | 0.03          |
|         | Peepers    | 1554           | 66.04     | 0.9999  | 0.03          |
|         | **Average** | **1640**       | **65.87** | **0.9999** | **0.03** |
| Kodak   | kodim04    | 1309           | 67.19     | 0.9999  | 0.03          |
|         | kodim05    | 1876           | 65.23     | 0.9999  | 0.04          |
|         | kodim15    | 1332           | 66.81     | 0.9999  | 0.03          |
|         | kodim23    | 1355           | 66.70     | 0.9999  | 0.03          |
|         | **Average** | **1468**       | **66.48** | **0.9999** | **0.03** |
| STARE   | im0005     | 1212           | 67.55     | 0.9999  | 0.02          |
|         | im0034     | 1253           | 66.95     | 0.9999  | 0.03          |
|         | im0086     | 1223           | 67.04     | 0.9999  | 0.02          |
|         | im0280     | 1292           | 67.12     | 0.9999  | 0.03          |
|         | **Average** | **1245**       | **67.16** | **0.9999** | **0.03** |
positions where secret data can be embedded is referred to as capacity. In the proposed scheme, $S$ blocks store 1-bit data, $M$ blocks store 2-bit data and $W$ blocks store 3-bit data in the appropriate channels. So, the embedding capacity can be calculated by Eq. (13). Similarly, the payload is referred to as bits per pixel (bpp), i.e., a number of secret bits embedded per pixel. For a designed data hiding scheme, Eq. (14) measures the payload of an image. The highest embedding capacity of 31858 is obtained for ucid00022 image of size $(512 \times 512)$ against PSNR 65.09 dB. The proposed scheme delivers a maximum payload of 0.05 bpp and a minimum of 0.02 bpp. On average, the payload reached is 0.03 bpp for the developed technique. The payload varies from image to image, mainly due to the generation of different numbers of $S$, $M$ and $W$ blocks.

$$\text{Capacity} = \left( \sum S \times 1 \right) + \left( \sum M \times 2 \right) + \left( \sum W \times 3 \right)$$

(13)

$$\text{Payload} = \frac{\text{Capacity}}{M \times N \times C} \text{bpp}$$

(14)

A graphical representation of the performance of the studied image databases is depicted in Fig. 11. The graph presents the average PSNR collected by the four image databases for the average payload achieved on a considered set of images. It can be observed that none of the databases shows any unexpected behavior during data hiding. With an increase in payload, the PSNR has decreased relatively less, which admits the significance of the proposed scheme. The UCID and USC-SIPI image databases have considerably produced a high payload compared to other databases.

Further, a comparison is drawn in Table 6 for PSNR and payload with state-of-the-art techniques present in the literature. This comparison is done only for the common color images of “Aeroplane,” “Baboon,” “Boat” and “Peepers.”
Table 6 Comparison of PSNR values with state-of-art schemes for the stego images

| Schemes          | Method                  | Aeroplane Payload | Baboon Payload | Boat Payload | Peepers Payload | Average Payload |
|------------------|-------------------------|-------------------|----------------|--------------|-----------------|-----------------|
| Su et al. (2019) | DCT + LSB               | 0.001             | 37.93          | 0.001        | 37.84           | 0.001           | 37.73           |
| Chowdhuri et al. (2018) | Weighted Matrix+DCT     | 0.07              | 40.41          | 0.06         | 40.90           | 0.07            | 40.67           | 0.07            | 40.21           |
| Yuan et al. (2020) | DCT + LSB               | 0.001             | 36.33          | 0.001        | 35.66           | 0.001           | 35.53           | 0.001           | 35.80           |
| Ashraf et al. (2020) | IT2FLS + LSB                | 0.25              | 51.41          | 0.25         | 51.12           | 0.25            | 51.36           | 0.25            | 51.58           |
| Sharma et al. (2021) | ABC + LWT-DCT             | 0.001             | 41.46          | 0.001        | 41.21           | 0.001           | 41.38           | 0.001           | 41.44           |
| Proposed        | T1FLS + LSB              | 0.03              | 67.07          | 0.04         | 65.44           | 0.03            | 66.50           | 0.02            | 67.27           | 0.03            | 66.57           |

Fig. 12 Graphical comparison of PSNR between existing schemes and the proposed scheme

matched with the mentioned recent schemes and follows LSB techniques for data embedding. A comparative study suggests that in terms of PSNR, the proposed scheme has gained high performance; however, the payload remains a concern. Additionally, a graphical comparison is presented in Fig. 12 to show the PSNR achieved by each state-of-the-art scheme for the compared images. It can be clearly noticed that none of the schemes provided many variations in PSNR with changes in images.

As suggested by Setiadi (2020) that SSIM is a better measure of imperceptibility in all aspects, Table 7 presents the values obtained for SSIM, NCC and BER for the stego images of size (512 × 512) under considered image databases. SSIM is another tool to assess the imperceptibility of the image. The calculation of SSIM considers three main factors, specifically luminance, contrast and structure of the image. The range of SSIM lies between $-1$ and $+1$. SSIM value close to 1 implies a high similarity of the stego image with the compared image in all three dimensions. NCC and BER point to the correlation and error in terms of actual change in bits in the stego image, respectively, compared to the cover image. For the Table 7, it is evident that average SSIM value obtained is 0.9999. This relates to the high imperceptibility maintained by the designed scheme even after a significant embedding of secret bits. It can also be verified with the NCC value of 0.9999 gained throughout the experiment. The BER achieved is between 0.00044 and 0.00083, which is quite appreciable compared to the number of bits embedded in the stego image. A comparison of SSIM values obtained from the proposed scheme with existing techniques is provided in Table 8. The proposed scheme has demonstrated better imperceptibility than all the compared schemes.

5.2 Robustness analysis

It is implicit that data communication media undergo some illegal alterations by unauthorized users, either intentionally or unintentionally. So, it becomes imperative for a data hiding scheme to tolerate such image processing attacks for security. The designed scheme has been tested with nine different distortions comprising seven unique attacks to test robustness and security. Performance of the stego image on the application of various attacks is shown in Fig. 13. The figure displays CI, STI, ESI, RSI, RCI, and comparison metrics PSNR, NCC and BER. It is apparent from the figure that the designed scheme has resisted attacks such as Salt & Peeper, constant average, copy-move forgery, cropping, contrast and opaqueness significantly. However, the designed scheme has some limitations for the rotation attack. The highest and lowest NCC values observed between SI and RSI are 0.9815 and 0.9049. The secret image was extracted from the tampered stego image with acceptable NCC and BER values. Moreover, it is also evident that the original cover image was recovered in all the cases.

Analyzing further, Table 9 provides a comparison of CI and RCI in terms of standard deviation (SD), correlation and SSIM values under different possible attacking conditions. It is observed from the table that at a high rate of tamper, the difference in SD is high as well, which is expected due to variations in the average pixel value of the image. Moreover, the
Table 7  SSIM, NCC and BER of (512 × 512) stego images under different databases

| Database | Image   | SSIM  | NCC  | BER  |
|----------|---------|-------|------|------|
| UCID     | ucid00022 | 0.9999 | 0.9999 | 0.00083 |
|          | ucid00075 | 0.9999 | 0.9999 | 0.00063 |
|          | ucid00102 | 0.9999 | 0.9999 | 0.00055 |
|          | ucid00646 | 0.9999 | 0.9999 | 0.00056 |
|          | Average  | 0.9999 | 0.9999 | 0.00064 |
| USC_SIPI | Aeroplane | 0.9999 | 0.9999 | 0.00051 |
|          | Baboon   | 0.9999 | 0.9999 | 0.00077 |
|          | Boat     | 0.9999 | 0.9999 | 0.00060 |
|          | Peepers  | 0.9999 | 0.9999 | 0.00050 |
|          | Average  | 0.9999 | 0.9999 | 0.00060 |
| Kodak    | kodim04  | 0.9999 | 0.9999 | 0.00049 |
|          | kodim05  | 0.9999 | 0.9999 | 0.00066 |
|          | kodim15  | 0.9999 | 0.9999 | 0.00046 |
|          | kodim23  | 0.9998 | 0.9999 | 0.00047 |
|          | Average  | 0.9999 | 0.9999 | 0.00052 |
| STARE    | im0005   | 0.9998 | 0.9999 | 0.00044 |
|          | im0034   | 0.9998 | 0.9999 | 0.00044 |
|          | im0086   | 0.9998 | 0.9999 | 0.00045 |
|          | im0280   | 0.9998 | 0.9999 | 0.00045 |
|          | Average  | 0.9998 | 0.9999 | 0.00045 |

Table 8  Comparison of SSIM values with state-of-art schemes for the stego images

| Schemes               | Aeroplane | Baboon | Boat | Peepers | Average |
|-----------------------|-----------|--------|------|---------|---------|
| Su et al. (2019)      | 0.9353    | 0.9794 | 0.9433 | 0.9231 | 0.9453  |
| Chowdhuri et al. (2018)| 0.9887   | 0.9872 | 0.9857 | 0.9879 | 0.9874  |
| Yuan et al. (2020)    | 0.9562    | 0.9854 | 0.9579 | 0.9382 | 0.9594  |
| Ashraf et al. (2020)  | 0.9955    | 0.9988 | 0.9974 | 0.9962 | 0.9968  |
| Sharma et al. (2021)  | 0.8994    | 0.9899 | 0.9790 | 0.9333 | 0.9661  |
| Proposed              | 0.9999    | 0.9999 | 0.9999 | 0.9999 | 0.9999  |

correlation and SSIM of the recovered cover image compared to the original cover image points to the fact that values are inversely proportional to the amount of distortion. The average correlation and SSIM obtained for the considered attacks are 0.8431 and 0.7454, respectively. These results uphold the effectiveness of the proposed scheme.

5.3 Self-recovery evaluation of secret image

At the time of embedding, copies of secret bits were embedded to counter the problem of tampering coincidence, so that secret bits are recovered properly after image processing attacks. During extraction, the secret image constructed with vector $O_1$ is ESI. However, post-processing is done with vectors $O_1$ and $O_2$ to recover a secret image (RSI) with greater visual quality and structure. Table 10 shows the enhancement achieved by RSI in terms of PSNR and SSIM. After post-processing, the recovered secret image has average visual quality of PSNR 2.40 dB higher than the extracted secret image. Furthermore, there is an approximately 17% increase in SSIM compared to earlier extracted secret images (ESI).

5.4 Steganalysis

Steganalysis is the method of detecting the presence of a hidden message in a suspected image. Steganalysis is less concerned with the existence of the covert message than with the traces left behind by data embedding. As a result, it is difficult to devise a data hiding scheme that is both imperceptible and resistant to these attacks. A steganalyst can perform this identification in various ways, but visual and statistical attacks are the most common. The efficacy of the built scheme
Fig. 13  Effects of proposed scheme under different attacks
Table 9  Comparison of cover image and recovered cover image under different attacking environment

| Types of Attack       | Rate of Tamper | SD of CI  | SD of RCI | Difference in SD | Correlation | SSIM  |
|-----------------------|----------------|-----------|-----------|------------------|-------------|-------|
| Salt & pepper         | 1              | 110.48    | 112.16    | 1.68             | 0.9771      | 0.8707|
| Salt & pepper         | 10             | 110.48    | 113.33    | 12.85            | 0.8034      | 0.3514|
| Salt & pepper         | 50             | 110.48    | 171.47    | 60.99            | 0.3253      | 0.0562|
| Constant average      | 10             | 110.48    | 114.34    | 3.86             | 0.8796      | 0.9801|
| Copy move forgery     | 10             | 110.48    | 110.02    | 1.46             | 0.9915      | 0.9838|
| Cropping              | 10             | 110.48    | 126.05    | 13.57            | 0.8374      | 0.9716|
| Contrast              | 10             | 110.48    | 121.60    | 8.12             | 1           | 0.9949|
| Opaque                | 10             | 110.48    | 110.47    | 4.01             | 1           | 0.9999|
| Rotation              | 1              | 110.48    | 123.75    | 12.27            | 0.7738      | 0.5001|
| Average               |                | 110.48    | 123.695   | 12.76            | 0.8431      | 0.7454|

Table 10  Recovery evaluation of secret image in terms of PSNR and SSIM

| Types of Attack       | Rate of Tamper | PSNR (ESI) | PSNR (RSI) | Difference in PSNR | SSIM (ESI) | SSIM (RSI) | Difference in SSIM |
|-----------------------|----------------|------------|------------|--------------------|-------------|-------------|--------------------|
| Salt & pepper         | 1              | 17.24      | 19.66      | 2.42               | 0.7024      | 0.7748      | 0.0724             |
| Salt & pepper         | 10             | 11.97      | 16.77      | 4.80               | 0.4387      | 0.9058      | 0.4672             |
| Salt & pepper         | 50             | 6.53       | 9.15       | 2.62               | 0.2052      | 0.5365      | 0.3313             |
| Constant average      | 10             | 9.73       | 11.53      | 1.80               | 0.2211      | 0.3210      | 0.0998             |
| Copy move forgery     | 10             | 18.66      | 20.36      | 1.70               | 0.9031      | 0.9422      | 0.0391             |
| Cropping              | 10             | 17.16      | 19.62      | 2.46               | 0.8956      | 0.9269      | 0.0313             |
| Contrast              | 10             | 8.65       | 9.97       | 1.32               | 0.1689      | 0.2228      | 0.0539             |
| Opaque                | 10             | 17.76      | 19.08      | 1.32               | 0.8505      | 0.9367      | 0.0862             |
| Rotation              | 1              | 5.14       | 8.24       | 3.10               | 0.0140      | 0.3619      | 0.3479             |
| Average               |                | 12.53      | 14.93      | 2.40               | 0.4888      | 0.6587      | 0.1699             |

is demonstrated in this paper using histogram attacks and RS analysis.

5.4.1 Histogram attack

To analyze the frequency distribution of the cover and stego “Baboon” picture, histogram attacks were chosen as one of the statistical attacks. Figure 14 depicts the frequency distribution of red, green and blue channels present in colored cover and stego image. It is clear from the comparison that the difference in the histograms of the two images is negligible. The consistency maintained for the amount of distortion induced by the embedding algorithm is evident. The obtained histograms do not display any sharp rise or variations that could reveal information on data hiding.

5.4.2 RS analysis

Fridrich et al. (2001) developed the RS steganalysis method used in this paper. This approach is highly effective against most forms of LSB steganography and it takes advantage of the fact that, contrary to common opinion, the bit planes of an image produced using LSB steganography methods are highly correlated. A discrimination function is applied to determine the noisiness of a group of pixels. Regular (R) and singular (S) are the groups where noise levels rise and fall, respectively. The flipping functions are M and -M. The theory is that regardless of which flipping function is used, the number of R and S groups in a standard image would be the same. The RS value is derived using Eq. (15).

\[
\frac{(|R_M - R_{-M}| + |S_M - S_{-M}|)}{(R_M + S_M)}
\]  

(15)

The experimental value obtained after RS steganalysis on four USC-SIPI stego images is provided in Table 11. From the data, it is ascertained that \(R_M \cong R_{-M}\) and \(S_M \cong S_{-M}\). The RS value for all the images is nearly equal to zero. This implies that the difference in a group of pixels is significantly less even after data embedding, which is imperative for a data hiding scheme with high security.
5.5 Comparison with Ashraf et al. (2020)

Compared to the technique demonstrated by Ashraf et al. (2020) using interval Type-2 Fuzzy and LSB substitution, the proposed scheme built using Type-1 Fuzzy and LSB substitution performed well in terms of visual quality and SSIM. The average visual quality achieved for the proposed scheme is 66.57 dB which is 14.99 dB higher than Ashraf et al. (2020). Furthermore, the SSIM procured for all the images is better than the compared scheme. But, the payload is limited to 0.03 bpp, which is relatively less than the embedding capacity of Ashraf et al. (2020). The developed scheme could resist non-geometrical attacks in addition to the ability to recover the confidential data even after tampering, which is not evident in the compared scheme. Due to the use of type-I fuzzy for
proximity calculation, the proposed technique’s time complexity is less and simpler.

6 Computational complexity

The computational complexity of the described protocol is $O(b^2 MN)$. According to Algorithm 1, it has three main subsections: line no. 3–6, line no. 7–16 and line no. 17–19. The first subsection has the time complexity of $O(3)$, while the next subsection has the complexity of $O(z \times b \times b \times b \times b)$. Finally, the time complexity of the last section is denoted by $O(z)$. So, the computation time

$$T(M, N) = O(z) + O(z \times b \times b \times b \times b) + O(z)$$

$$= O(z \times b^4)$$

$$= O \left( \frac{M \times N}{b^2 \times b^4} \right), \text{ since, } z = \frac{M \times N}{b \times b}$$

$$= O(b^2 MN)$$

7 Conclusion

This paper proposes a robust scheme for data hiding by incorporating fuzzy logic and interval threshold. The designed scheme starts with block segregation followed by proximity calculation using Mamdani rule-based fuzzy logic. Based on block proximity and interval threshold, a block is graded strong, moderate or weak where secret data are hidden. A secret bit is replicated at two random blocks and channels generated through a shared secret key to recover the secret data with high visual quality. To show the efficacy of the proposed scheme, various experiments, analyses and comparisons are made. The perceptual transparency and visual quality of the stego image were measured by PSNR, Q-Index and SSIM. Experimental results have justified that the designed scheme maintained high PSNR, SSIM and Q-Index. On average, the proposed scheme exhibited PSNR of 95 dB, payload 0.03 and SSIM 0.9999. The suggested scheme could also resist different image processing attacks, with a significant amount of recovery of the secret images. As the payload capacity of the proposed technique is relatively low, the future study will focus on increasing the embedding capacity and resilience to various geometrical attacks. Few shortcomings that exist in terms of the visual quality of the recovered secret image shall also be looked upon to overcome.

Funding We declare this work is an independent work and no financial assistance has been received for the work.

Data Availability The data that support the findings of this study are available from the corresponding author upon reasonable request.

Declaration

Conflict of interest We declare we have no conflict of interest.

References

Abraham A, Paprzycki M, et al (2004) Significance of steganography on data security. In: International conference on information technology: coding and computing, 2004. Proceedings. ITCC 2004, vol 2. IEEE, pp 347–351

Ashraf Z, Roy ML, Muhuri PK, Lohani QD (2020) Interval type-2 fuzzy logic system based similarity evaluation for image steganography. Helijon 6(5):e03771

Atawneh S, Almomani A, Al Bazar H, Sumari P, Gupta B (2017) Secure and imperceptible digital image steganographic algorithm based on diamond encoding in DWT domain. Multimed Tool Appl 76(18):18451–18472

Balasubramanian C, Selvakumar S, Geetha S (2014) High payload image steganography with reduced distortion using octonary pixel pairing scheme. Multimed Tools Appl 73(3):2223–2245

Bhaumik A, Roy SK, Weber GW (2020) Hesitant interval-valued intuitionistic fuzzy-linguistic term set approach in prisoners’ dilemma game theory using topsis: a case study on human-trafficking. Cent Eur J Oper Res 28(2):797–816

Chandramouli R, Memon N (2001) Analysis of LSB image steganography techniques. In: Proceedings 2001 international conference on image processing

Chang CC, Yu YH, Hu YC (2008) Hiding secret data into an ambient-compressed image using genetic algorithm. In: 2008 second international conference on future generation communication and networking symposia, vol 3. IEEE, pp 154–157

Chowdhuri P, Jana B, Giri D (2018) Secured steganographic scheme for highly compressed color image using weighted matrix through dct. Int J Comput Appl 43:1–12

Dadgostar H, Afsari F (2016) Image steganography based on interval-valued intuitionistic fuzzy edge detection and modified lsb. J Inf Secur Appl 30:94–104

Demirici R (2006) Rule-based automatic segmentation of color images. AEU Int J Electron Commun 60(6):435–442

Dubois DJ (1980) Fuzzy sets and systems: theory and applications, vol 1. Academic press, Boston

Fkirin A, Attiya G, El-Sayed A (2021) Two-level security approach combining watermarking and encryption for securing critical colored images. Opt Quant Electron 53(6):1–27

Fridrich J, Goljan M, Du R (2001) Invertible authentication. In: Security and watermarking of multimedia contents III, vol 4314. International Society for Optics and Photonics, pp 197–208

Haibo SJXHH (2008) Discrete fourier transform-based information steganography. J Huazhong Univ Sci Technol Nat Sci Ed 36:08

Hamid N, Yahya A, Ahmad RB, Al-Qershi OM (2012) A comparison between using sift and surf for characteristic region based image steganography. Int J Comput Sci Issues IJCSI 9(3):110

Hamid N, Yahya A, Ahmad R B, Al-Qershi O (2012a) Characteristic region based image steganography using speeded-up robust features technique. In: 2012 International conference on future communication networks, IEEE, pp 141–146

Jagadeesh B, Kumar PR, Reddy PC (2016) Robust digital image watermarking based on fuzzy inference system and back propagation neural networks using dct. Soft Comput 20(9):3679–3686

Jia Y, Yin Z, Zhang X, Luo Y (2019) Reversible data hiding based on reducing invalid shifting of pixels in histogram shifting. Signal Process 163:238–246

Springer
Khashandarag AS, Navin AH, Mirmia MK, Mohammadi HHA (2011) An optimized color image steganography using lfsr and dft techniques. In: International conference on computer education, simulation and modeling, Springer, pp 247–253

Kiani S, Moghaddam ME (2009) Fractal based digital image watermarking using fuzzy c-mean clustering. In: 2009 international conference on information management and engineering, IEEE, pp 638–642

Lee T-Y, Lin SD (2008) Dual watermark for image tamper detection and recovery. Pattern Recogn 41(11):3497–3506

Lee C-F, Weng C-Y, Kao C-Y (2019) Reversible data hiding using lagrange interpolation for prediction-error expansion embedding. Soft Comput 23(19):9719–9731

Maity SP, Kundu MK (2009) Genetic algorithms for optimality of data hiding in digital images. Soft Comput 13(4):361–373

Molina-Garcia J, Garcia-Salgado BP, Ponomaryov V, Reyes-Reyes R, Sadovnychiy S, Cruz-Ramos C (2020) An effective fragile watermarking scheme for color image tampering detection and self-recovery. Signal Process Image Commun 81:115725

Mungmode S, Sedamkar R, Kulkarni N (2016) A modified high frequency adaptive security approach using steganography for region selection based on threshold value. Procedia Comput Sci 79:912–921

Niimi M, Noda H, Kawaguchi E (1999) Steganography based on region segmentation with a complexity measure. Syst Comput Jpn 30(3):1–9

Patel N, Meena S (2016) Lsb based image steganography using dynamic key cryptography. In: 2016 international conference on emerging trends in communication technologies (ETCT), IEEE, pp 1–5

Rajendran S, Doraipandian M (2017) Chaotic map based random image steganography using lsb technique. J Netw Secur 19(4):593–598

Roy SK, Bhamuik A (2018) Intelligent water management: a triangular type-2 intuitionistic fuzzy matrix games approach. Water Resour Manag 32(3):949–968

Sajedi H, Jamzad M (2008) Cover selection steganography method based on similarity of image blocks. In: 2008 IEEE 8th international conference on computer and information technology workshops, IEEE, pp 379–384

Seki Y, Kobayashi H, Fujiyoshi M, Kiya H (2005) Quantization-based image steganography without data hiding position memorization. In: 2005 IEEE international symposium on circuits and systems. IEEE, pp 4987–4990

Setiadi DRIM (2020) Psnr vs ssim: imperceptibility quality assessment for image steganography. Multimed Tools Appl

Shaﬁ I, Noman M, Gohar M, Ahmad A, Khan M, Din S, Ahmad SH, Ahmad J (2018) An adaptive hybrid fuzzy-wavelet approach for image steganography using bit reduction and pixel adjustment. Soft Comput 22(5):1555–1567

Sharma S, Sharma H, Sharma JB (2021) Artificial bee colony based perceptually tuned blind color image watermarking in hybrid LWT-DCT domain. Multimed Tools Appl 80:1–33

STARE (2017) University of california, san diego. stare image database. https://cecas.clemson.edu/ahoover/stare/. Accessed September 20, 2017

Su Q, Liu D, Yuan Z, Wang G, Zhang X, Chen B, Yao T (2019) New rapid and robust color image watermarking technique in spatial domain. IEEE Access 7:30398–30409

Sutaone M, Khandare M (2008) Image based steganography using LSB insertion

Tang L, Wu D, Wang H, Chen M, Xie J (2021) An adaptive fuzzy inference approach for color image steganography. Soft Comput 25(16):10987–11004

Tang L, Xie J, Wu D (2022) An interval type-2 fuzzy edge detection and matrix coding approach for color image adaptive steganography. Multim Tools Appl 1–23

UCID (2017) Nottingham trent university, ucid image database. http://sipi.usc.edu/database/database.php. Accessed September 20, 2017

USC-SIPI (2017) University of southern california. the usc-sipi image database. http://sipi.usc.edu/database/database.php. Accessed September 20, 2017

Valandar MY, Barani MJ, Ayubi P (2020) A blind and robust color images watermarking method based on block transform and secured by modified 3-dimensional hénon map. Soft Comput 24(2):771–794

Wong A, Ho S, Olusanya O, Antonini MV, Lyness D (2021) The use of social media and online communications in times of pandemic covid-19. J Intensiv Care Soc 22(3):255–260

Wuerger SM, Maloney LT, Krauskopf J (1995) Proximity judgments in color space: tests of a euclidean color geometry. Vis Res 35(6):827–835

Yang CH, Weng CY, Chen JY (2022) High-fidelity reversible data hiding in encrypted image based on difference-preserving encryption. Soft Comput 26(4):1727–1742

Yuan Z, Liu D, Zhang X, Wang H, Su Q (2020) DCT-based color digital image blind watermarking method with variable steps. Multimed Tools Appl 79(41):30557–30581

Zadeh L (1965) Fuzzy sets. Inf Control 8(3):338–353

Zadeh LA (1978) Fuzzy sets and their extension to a theory of possibility. Fuzzy Sets Syst 1(1):3–28

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.