Fuzzy Proximity based Robust Data Hiding Scheme with Interval Threshold

Prabhas Kumar Singh
Vidyasagar University
Biswapati Jana (biswapatijana@gmail.com)
Vidyasagar University https://orcid.org/0000-0003-4476-3459
Kakali Datta
Visva-Bharati University: Visva-Bharati

Research Article

Keywords: Data hiding , Fuzzy logic , Proximity , Tampering , Steganalysis

Posted Date: January 5th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-563015/v1

Noname manuscript No.
(will be inserted by the editor)

Fuzzy Proximity based Robust Data Hiding Scheme with Interval Threshold

Prabhash Kumar Singh · Biswapati Jana* · Kakali Datta

the date of receipt and acceptance should be inserted later

Abstract In 2020, Ashraf et al. proposed an interval type-2 fuzzy logic based block similarity calculation using color proximity relations of neighboring pixels in a steganographic scheme. Their method works well for detecting similarity, but it has drawbacks in terms of visual quality, imperceptibility, security, and robustness. Using Mamdani fuzzy logic to identify color proximity at the block level, as well as a shared secret key and post-processing system, this paper attempts to develop a robust data hiding scheme with similarity measure to ensure good visual quality, robustness, imperceptibility, and enhance the security. Further, the block color proximity is graded using an interval threshold. Accordingly, data embedding is processed in the sequence generated by the shared secret keys. In order to increase the quality and accuracy of the recovered secret message, the tampering coincidence problem is solved through a post-processing approach. 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

Prabhash Kumar Singh
Department of Computer Science, Vidyasagar University, West Midnapore, West Bengal, India
E-mail: singhg11@gmail.com

{*Corresponding Author} Biswapati Jana
Department of Computer Science, Vidyasagar University, West Midnapore, West Bengal, India
E-mail: biswapatijana@gmail.com

Kakali Datta
Department of Computer and System Sciences, Visva-Bharati University, Santiniketan, India
E-mail: kakali.datta@visva-bharati.ac.in

1 Introduction

The COVID-19 pandemic has made us realize the importance of the internet and multimedia technologies. The complete communication system all over the world has moved to the internet using multimedia data. They have been the actual source of data transmission and information exchange. 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 be used to modify digital multimedia documents quickly and professionally. This has aided in the creation of various image forgery attacks. Authentication has been important for tamper detection and/or owner identification in many human centric applications, including forensic analysis, insurance processing, surveillance systems, radiography imaging, journalism and others. In this sense, 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 is the soul of data communication. Complete data hiding systems can be put into two categories, one is information hiding and another is encryption. Generally, watermarking is applied to hide or embed a message in a cover image so that authenticity and copyright could be proved when asked for. The watermark applied can be visible or invisible. Steganography, on the other hand, 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 a

variety of purposes, including secret data sharing, secure correspondence and authentication. Alternatively, 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 to feature better in robustness, imperceptibility, security and hiding capacity. However, at any point of time, it is arduous to extract maximum out of each mentioned features. There has to be a pareto-optimal solutions so that each properties are maintained at its optimum level. An increase in the amount of one would lead to the decrease of the other. Also, at times, the development of a data hiding scheme is lead considering the field of application. In some field, robustness is preferred while some gives priority to high security and imperceptibility. As suggested by (Abraham et al., 2004), robustness is generally required for a system where a chance of unwanted malicious attacks are high. Since within watermarking, a watermark can be either visible or invisible, therefore a high visual quality or capacity is not always required.

In this paper, an emphasis has been put to design 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 secret message at random blocks. As a result, even after tampering of 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 the deletion of a specific region and its recovery 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.

i) Cover Image: The original image used for embedding of secret message.
ii) Message: Secret data that is sent by the sender to the receiver.
iii) Stego Image: The image obtained after hiding the message.
iv) Robustness: The capability of an image to resist attacks.
v) Imperceptibility: The strength of a steganographic technique to hide the message in a way that is not identifiable by a naked eye or image statistic.

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

2 Related Work

The relevant investigation regarding hidden data communication has progressed over time. In the literature, there are many data-hiding-based authentication methods has been reported by many researchers in recent time. Authentication methods are distinguished by three main characteristics: 1) data generation for authentication, 2) data generation for recovery and 3) the data embedding process used. In multimedia authentication, fragile data hiding scheme incorporating Least Significant Bit (LSB) replacement are commonly used. For any data hiding scheme, imperceptibility is the major concern. The message must be embedded in such a way that its presence is not tracked either statistically or visually (Cox et al., 2007).

Table 1: List of notations

Notations	Descriptions
Δ_{RGB}	Difference in Intensity (colordiff)
ψ	Euclidean Distance (closeness)
α	Bias Weight
R_i	i^{th} Fuzzy Rule
χ	Proximity of Pixel
χ_{cpr}	Cumulative Proximity of Pixel
ξ	Block Proximity
$[t^-, t^+]$	Interval Threshold
ϵ	Degree of Elasticity
$[t^- + \epsilon, t^+ + \epsilon]$	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

In a steganography, the information required for communication to the receiver is embedded in the cover image by a sender. On receiving, a receiver segregates the hidden data. There are two types of motivations that can be used to design a steganographic process: reversible and irreversible. In a 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 is inserted directly in 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 maximise payload capability by using the majority of the cover image pixels. The research issue became more oriented after that, with the goal of developing 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 optimise 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 of 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 an image steganographic 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). In comparison 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. 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). Researchers experimented with various ways to improve imperceptibility in the steganographic system. 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). Chang et al. (2008) used a 2-level quantization method along with Genetic Algorithm in which three bits are additionally embedded over image blocks with high fidelity to try to maximise payload power. As a result, the Genetic Algorithm is extremely resistant to image manipulation. However, finding the local optimal solution requires a large number of computational stages, which slows down the search for specific embedding locations.

At the expense of increased modeling complexity, a fuzzy logic-based method focuses more on visual consistency preservation to enhance stego-media imperceptibility. 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 spec-

ify the semantic rules for the embedding process using these features from cover image sub-regions. Also at higher embedding rate, the principle helps to minimise stego image distortions. Few recent investigator (Jagadeesh et al., 2016; Tang et al., 2021) 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 a variety of 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.

From the overview of the aforementioned discussion, 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 in such a way that all elements are categorise 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 function by the following expression:

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

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 makes 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 $x \in X$ be an element present in a fuzzy set $\tilde{A}$. The degree of membership of x in $\tilde{A}$ is indicated by $\mu_{\tilde{A}}(x)$. Subsequently, the fuzzy set is represented in terms of ordered pair $\tilde{A} = (x, \mu_{\tilde{A}}(x)/x \in X)$.

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 is convex and continuous in nature.

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

$$pos(\tilde{A_1} * \tilde{A_2}) = sup\{min(\mu_{\tilde{A_1}}(x), \mu_{\tilde{A_2}}(y)), x, y \in \Re, x * y\} \quad (2)$$

where *pos* represents possibility, $*$ is any one of the relations $>, <, =, \leq, \geq$ 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 $\tilde{A}$ is continuous or discrete. A continuous fuzzy number can be represented by a special class of triangular fuzzy numbers, trapezoidal fuzzy numbers and Gaussian fuzzy numbers.

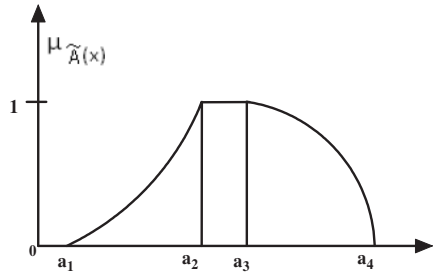

Fig. 1: General representation of a fuzzy number

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

$$\mu_{\tilde{A}}(x) = \begin{cases} \frac{x-a_1}{a_2-a_1} & \text{for } a_1 \leq x \leq a_2 \\ \frac{a_3-x}{a_3-a_2} & \text{for } a_2 \leq x \leq a_3 \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

Linguistic variables: The concept of linguistic variables is crucial to the growth of fuzzy set theory. 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

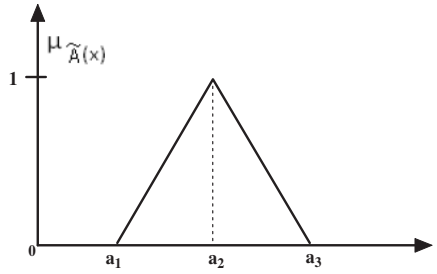

Fig. 2: Triangular fuzzy number representation

denoted as a linguistic variable such as very hot, hot, cold, very cold and so on.

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:

If X_1 is $\tilde{A_1}$ and ... X_n is $\tilde{A_n}$ then Y is B. (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. Fig. 3 clearly 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 of these channels 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 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 tends to follow the law of proximity as defined in the Gestalt grouping law. This simple rule states that objects that are close together are more likely to be grouped together, than the objects that are further apart. Thus, in the proposed research, while calculating the proximity of pixel (χ), importance has been given to the difference in intensity (color) and distance (closeness) as well.

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

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

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

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 equation (5). Here, $\Delta_{RGB}(P_i, P_j)$ represents the difference in intensity values of red, green and blue channels as provided in equation (6). The closeness of two pixels are measured in terms of Euclidean distance between them through equation (7). The parameter α 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 it 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 as input for fuzzification. Each factor is represented as a fuzzy set with fuzzy linguistic variables as $low, medium$ and $high$. There are several membership functions to model these linguistic terms such as trapezoidal, Gaussian, triangular, etc. however, the triangular fuzzy number is chosen for the representation of the stated linguistic terms due to its simplicity and symmetric representation. According to equation (6), the minimum and maximum value that it can hold is 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.

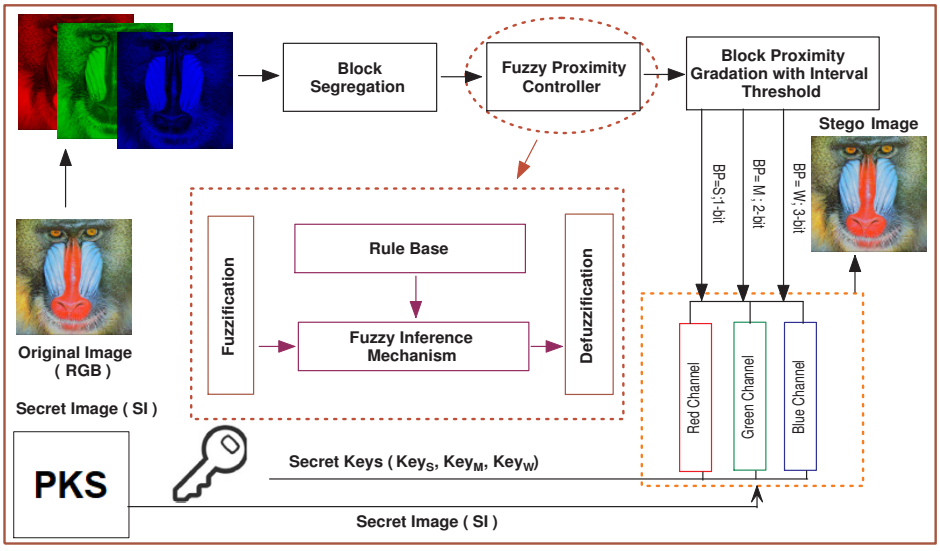

Fig. 3: Flowchart of the stages in the proposed embedding technique

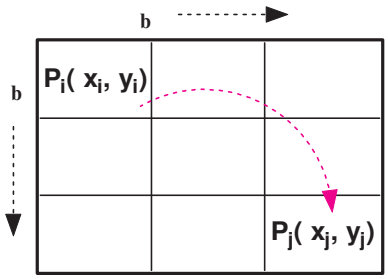

Fig. 4: Non-overlapping block of size $(b \times b)$

Table 2: Fuzzy Rules

Rule	Antecedent	Consequent	
R_1	If colordiff is *Low* and closeness is *High*	then	χ is *High*
R_2	If colordiff is *Low* and closeness is *Medium*	then	χ is *Medium*
R_3	If colordiff is *Low* and closeness is *Low*	then	χ is *Low*
R_4	If colordiff is *Medium* and closeness is *High*	then	χ is *High*
R_5	If colordiff is *Medium* and closeness is *Medium*	then	χ is *Medium*
R_6	If colordiff is *Medium* and closeness is *Low*	then	χ is *Low*
R_7	If colordiff is *High* and closeness is *High*	then	χ is *Medium*
R_8	If colordiff is *High* and closeness is *Medium*	then	χ is *Low*
R_9	If colordiff is *High* and closeness is *Low*	then	χ is *Low*

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 exist. 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 equation (4) is represented as:
R_i : If x_1 is $\tilde{A}_1, x_2$ is $\tilde{A}_2, ..., x_n$ is $\tilde{A}_n$ Then y is $\tilde{B}$
The set of inputs in the rule base are 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 χ which is denoted by the linguistic term as $Low, Medium$ 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 maximum.

The actions of individual rule are superimposed to construct a fuzzy output set as the final production. The fuzzy output is generated by combining the outputs of the rules that have been shot. On the fuzzy output, subsequent defuzzification methods generate a crisp value. Maxima methods and area-based methods are two popular defuzzification techniques. The centroid method is a common area-based defuzzification technique. The crisp value is the point of the output membership function that divides the region in half, as the name implies. In this way, for each P_i present in a block, we calculate the proximity pixels $\chi(P_i, P_j)$ such that $j = 1, 2, ..., b^2; j \neq i$. The proximity calculation network of pixels in a block are demonstrated in Fig. 7. At last, the cumulative proximity of pixel $\chi_{cpr}(P_i)$ is computed by equation (8).

$$\chi_{cpr}(P_i) = \frac{\sum_{j=1}^{b^2} \chi(P_i, P_j)}{b^2 - 1}; j \neq i \tag{8}$$

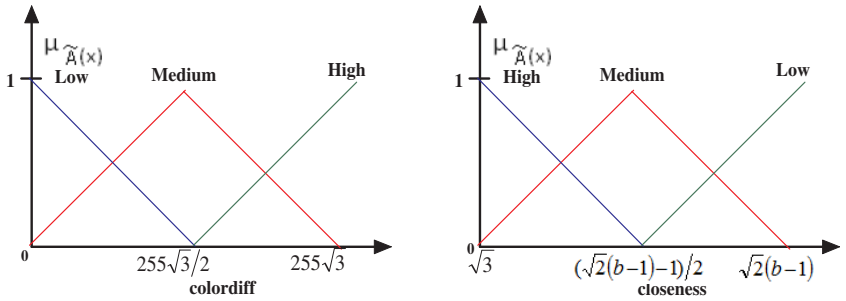

Fig. 5: Membership functions for fuzzy set colordiff and closeness

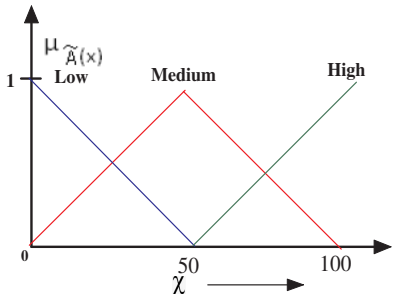

Fig. 6: Membership functions for proximity of pixels (prpix)

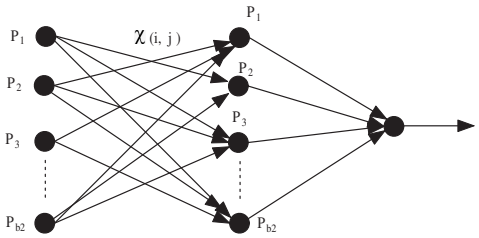

Fig. 7: Proximity calculation network

Now, the proximity of the block (ξ) in whole is determined by equation (9) as follows:

$$\xi = \frac{1}{b^2}\sum_{i=1}^{b^2}\chi_{cpr}(P_i) \tag{9}$$

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

4.3 Block Proximity Gradation with Interval Threshold

After calculation of block proximity (ξ), the blocks are graded as *strong* (S), *moderate* (M) and *weak* (W) based on proximity value, ξ, using interval threshold. In 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 ϵ is introduced to extend a degree of elasticity within the interval threshold so that threshold value t^- and t^+ does not behave rigidly. The value of ϵ may vary from image to image. Now, the interval threshold becomes $t = [t^- + \epsilon, t^+ + \epsilon]$. Before categorisation, the ξ of the blocks are normalized between 0 and 1. A $block(i)$ is graded as given in equation (10) and represented in Fig. 8.

$$block(i) = \begin{cases} S & \text{if } \xi \leq (t^- + \epsilon) \\ W & \text{if } \xi \geq (t^+ + \epsilon) \\ M & \text{otherwise} \end{cases} \tag{10}$$

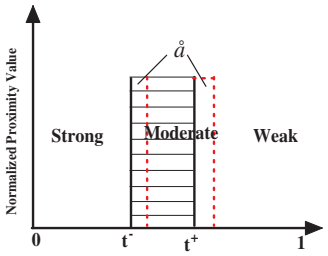

Fig. 8: Proximity calculation network

4.4 Data Embedding

For data embedding, similar graded blocks are grouped to hide data in the LSB. Blocks categorised as *strong*,

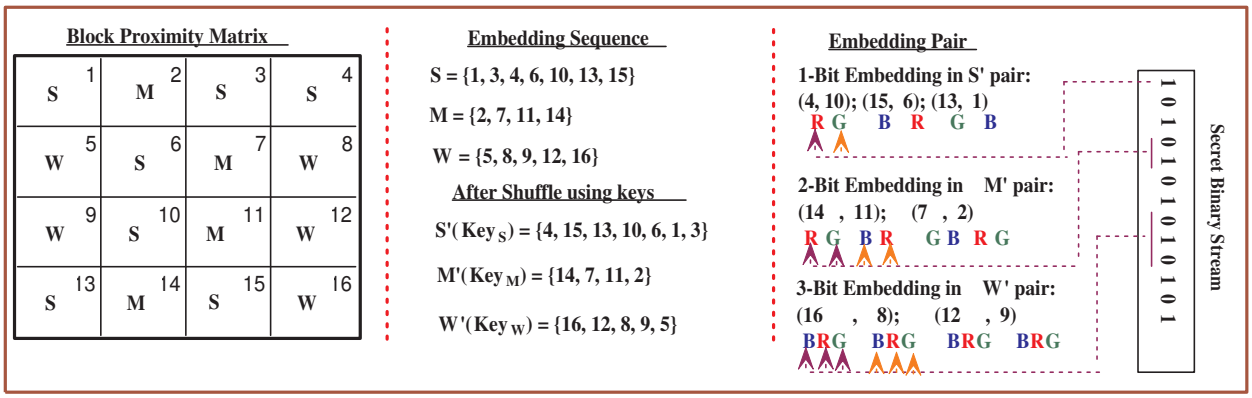

Fig. 9: Example of Data Embedding

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 bit stream. Each bit of the secret image is embedded twice in the appropriate channels of two different blocks specified through secret keys to overcome the problem of tampering coincidence (Molina-Garcia et al., 2020). This helps to recover the secret hidden image, after tampering with better accuracy and visual quality.

Let us consider an example shown in Fig. 9(i) where 16 blocks are graded as S, M or W. Accordingly, the blocks are grouped together as per assigned gradation. Thereafter, corresponding secret keys are used to shuffle the sequence of blocks in respective groups (Fig. 9(ii)). New sequence obtained are marked as $S^{'}(Key_S)$, $M^{'}(Key_M)$ and $W^{'}(Key_W)$. Finally, the secret data bits are double embedded in pairs, starting from 1-bit insertion to 3-bit insertion. Also, the order of insertion is maintained in color channels too. As given in the Fig. 9(iii), the first secret bit 1 is embedded in red channel of 4^{th} block and green channel of 10^{th} block. Second bit, 0 is embedded in blue and red channel of 15^{th} and 6^{th} blocks respectively. Further, 2-bit embedding $(0, 1)$ is carried out in two channels such as red and green channel of blocks 14^{th} and 11^{th}. Similarly, 3-bit embedding $(0, 1, 0)$ is done in all three channels of block, but in sequence as appears, during data hiding such as blue, red and green channel of block 16^{th} and 8^{th}. Thus, in this way total data embedding of secret data is completed and finally, stego image is generated. A pseudocode of the given technique is provided in Algorithm 1.

Algorithm 1: Data embedding within color cover image.

input : Cover Image ($CI_{M\times N}$), Secret Image ($SI_{m\times n}$), Key(Key_S,Key_M,Key_W), t^+, t^-
output: Stego Images ($STI_{M\times N}$)

Algorithm DataEmbedding():
- I = ReadColorImage($CI_{M\times N}$);
- **for** $(x = 1; x <= 3; x++)$ **do**
  - CC_x = I(:,:,x);
  - B_x = BlockSegregation(b, b, CC_x);
  - // $(b \times b)$ is the block size
- **end**
- // z = total blocks of size $(b \times b)$
- **for** $(x = 1; x <= z; x++)$ **do**
  - **for** $(j = 1; j <= b; j++)$ **do**
    - **for** $(i = 1; i <= b; i++)$ **do**
      - $FP(i, j)$ = FuzzyProximity(B_x, i, j);
      - $tFP(i, j) = tFP(i, j) + FP(i, j)$;
      - // $tFP(i, j)$ = total Fuzzy Proximity
    - **end**
  - **end**
  - $CPR(x) = mean(tFP)$;
  - $BPR(x)$ = IntervalThreshold($CPR(x)$, t^+, t^-);
- **end**
- **for** $(x = 1; x <= z; x++)$ **do**
  - EmbedSecretBits(B, $SI_{m\times n}$);
- **end**
- CreateStegoImage();

Function FuzzyProximity(B_x, i, j):
- **for** $(m = 1; m <= b; m++)$ **do**
  - **for** $(n = 1; n <= b; n++)$ **do**
    - $prpix$ = PixelProximity($P(i, j)$, $P(m, n)$);
    - $tprpix = tprpix + prpix$;
    - // $tprpix$ = total pixel proximity
  - **end**
- **end**
- return $tprpix/(b^2 - 1)$;

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. The extraction of data 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 database (STARE, 2017). Out of each database, four popular images of size (512×512) are chosen to create a variation and unbiasedness during comparison (Fig. 10). This will help to assess the proposed scheme in a different environment with appropriate parameters and features. The experimental simulations are conducted in MATLAB on a Windows platform with 4GB RAM and a 2.6 GHz Intel core i5 processor. 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.

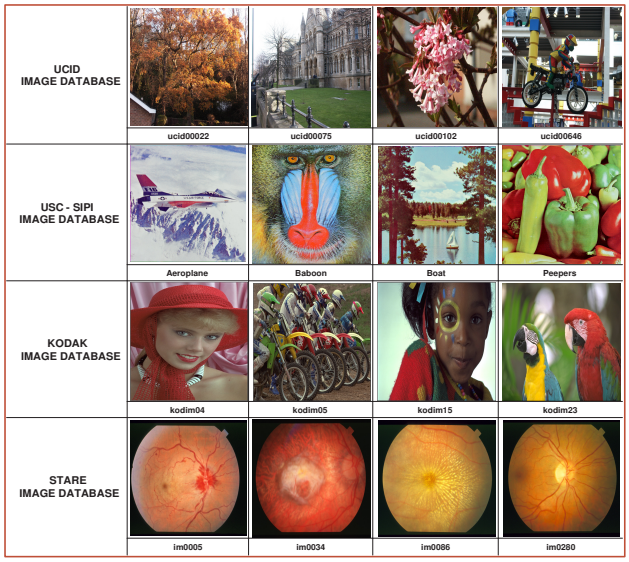

Fig. 10: Set of considered cover images from standard databases for experiment

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 equations (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, used to judge distortion caused relative to three combination factors: loss of luminance distortion, contrast distortion and correlation.

$$MSE = \frac{1}{M \times N \times C} \sum_{i=1}^{M} \sum_{j=1}^{N} \sum_{k=1}^{C} (I_c(i,j,k) - I_s(i,j,k))^2 \tag{11}$$

$$PSNR = 10log_{10}(\frac{MAX^2}{MSE}) \tag{12}$$

To measure the quality and payload of the stego image, a precise analysis has been conducted on three different sizes of the cover image: (64×64), (128×128) and (512×512). The results obtained are revealed in Table 3 - 5. On observation, it can be stated that in all cases the PSNR value lies between 64 dB to 68 dB and Q-Index maintaining a constant 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, Table 3 - 5 shows the capability and payload achieved in each of the three cases described. The number of 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 equation (13). Similarly, the payload is referred to in terms of bits per pixel (bpp) i.e. number of secret bits embedded per pixel. For a designed data hiding scheme, equation (14) measures the payload of an image. The highest embedding capacity of 31858 is obtained for ucid00022 image of size (512×512) against PSNR 65.09 dB. The

Table 3: Capacity, PSNR, Q-Index and Payload values of (64×64) stego images

Dataset	Image	Capacity	PSNR	Q-Index	Payload
UCID	ucid00022	441	65.205	0.9999	0.04
	ucid00075	486	65.523	0.9999	0.04
	ucid00102	543	64.431	0.9999	0.04
	ucid00646	613	64.042	0.9999	0.05
	Average	521	64.80	0.9999	0.04
USC_SIPI	Aeroplane	435	65.621	0.9999	0.04
	Baboon	409	66.037	0.9999	0.03
	Boat	524	65.011	0.9999	0.04
	Peepers	481	65.641	0.9999	0.04
	Average	462	65.58	0.9999	0.04
Kodak	kodim04	382	65.787	0.9999	0.03
	kodim05	495	65.241	0.9999	0.04
	kodim15	387	66.307	0.9999	0.03
	kodim23	405	65.993	0.9999	0.03
	Average	417	65.83	0.9999	0.03
STARE	im0005	364	66.957	0.9999	0.03
	im0034	402	66.058	0.9999	0.03
	im0086	331	66.930	0.9999	0.03
	im0280	358	66.377	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	PSNR	Q-Index	Payload
UCID	ucid00022	1942	65.370	0.9999	0.04
	ucid00075	1860	65.540	0.9999	0.04
	ucid00102	1858	65.361	0.9999	0.04
	ucid00646	2006	65.089	0.9999	0.04
	Average	1917	65.34	0.9999	0.04
USC_SIPI	Aeroplane	1559	66.114	0.9999	0.03
	Baboon	1767	65.460	0.9999	0.04
	Boat	1680	65.855	0.9999	0.03
	Peepers	1554	66.042	0.9999	0.03
	Average	1640	65.87	0.9999	0.03
Kodak	kodim04	1309	67.192	0.9999	0.03
	kodim05	1876	65.228	0.9999	0.04
	kodim15	1332	66.811	0.9999	0.03
	kodim23	1355	66.700	0.9999	0.03
	Average	1468	66.48	0.9999	0.03
STARE	im0005	1212	67.550	0.9999	0.02
	im0034	1253	66.950	0.9999	0.03
	im0086	1223	67.039	0.9999	0.02
	im0280	1292	67.115	0.9999	0.03
	Average	1245	67.16	0.9999	0.03

Table 5: Capacity, PSNR, Q-Index and Payload values of (512×512) stego images

Dataset	Image	Capacity	PSNR	Q-Index	Payload
UCID	ucid00022	31858	65.087	0.9999	0.04
	ucid00075	25050	66.302	0.9999	0.03
	ucid00102	20760	66.907	0.9999	0.03
	ucid00646	21700	66.831	0.9999	0.03
	Average	24842	66.282	0.9999	0.03
USC_SIPI	Aeroplane	19832	67.070	0.9999	0.03
	Baboon	29333	65.442	0.9999	0.04
	Boat	22850	66.500	0.9999	0.03
	Peepers	18805	67.266	0.9999	0.02
	Average	22705	66.570	0.9999	0.03
Kodak	kodim04	18478	67.391	0.9999	0.02
	kodim05	25233	66.091	0.9999	0.03
	kodim15	18877	67.610	0.9999	0.02
	kodim23	18050	67.569	0.9999	0.02
	Average	20160	67.165	0.9999	0.02
STARE	im0005	16842	67.803	0.9999	0.02
	im0034	16906	67.820	0.9999	0.02
	im0086	17182	67.750	0.9999	0.02
	im0280	17109	67.753	0.9999	0.02
	Average	17010	67.782	0.9999	0.02

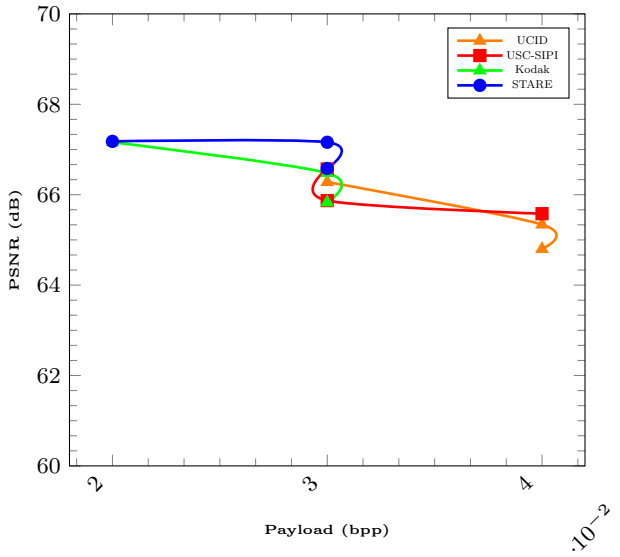

Fig. 11: Performance of image databases in terms of PSNR and Payload for different image sizes

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.

$$Capacity = (\sum S \times 1) + (\sum M \times 2) + (\sum W \times 3) \quad (13)$$

$$Payload(bpp) = \frac{Capacity}{M \times N \times C} \quad (14)$$

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 unanticipated behaviour during the course of 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 database have considerably produced a high payload in comparison 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

Table 6: Comparison of PSNR values with state-of-art schemes for the stego images.

Schemes	Method	Aeroplane		Baboon		Boat		Peepers		Average	
		Payload	PSNR	Payload	PSNR	Payload	PSNR	Payload	PSNR	Payload	PSNR
Su et al. (2019)	DCT + LSB	0.001	37.93	0.001	37.84	0.001	37.46	0.001	37.69	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	0.07	40.55
Yuan et al. (2020)	DCT + LSB	0.001	36.33	0.001	35.66	0.001	35.68	0.001	35.53	0.001	35.80
Ashraf et al. (2020)	IT2FLS + LSB	0.25	51.41	0.25	52.12	0.25	51.42	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	41.71	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

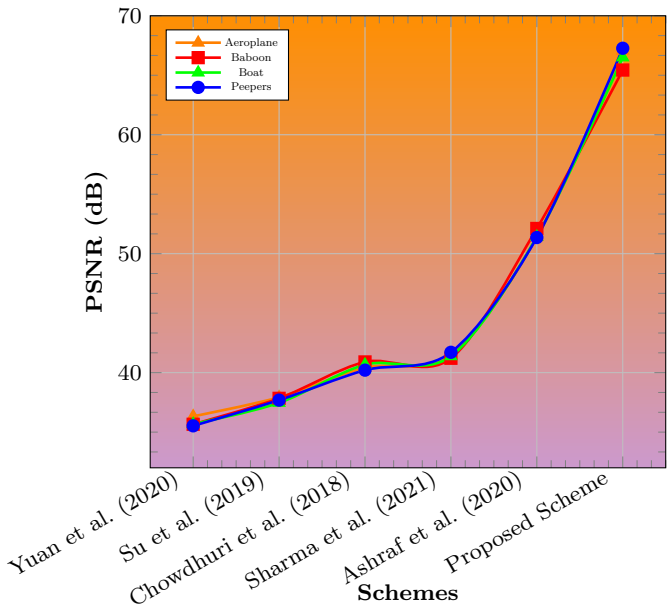

Fig. 12: Graphical comparison of PSNR between existing schemes and the proposed scheme

common color images of "Aeroplane", "Baboon", "Boat" and "Peepers" 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 change 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

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

of SSIM lies between -1 to $+1$. 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 stego image respectively, as 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 amount of embedding of secret bits. It can also be verified with NCC value

of 0.9999 gained throughout the experiment. The BER achieved is between 0.00044 and 0.00083 which is quite appreciable as compared to the number of bits embedded in the stego image. A comparison of SSIM values obtained from the proposed scheme with existing techniques are provided in Table 8. The proposed scheme has demonstrated better imperceptibility than all the compared schemes.

5.2 Robustness Analysis

It is implicit for a data communication media to 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 of 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 along with 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 opaque significantly. However, the designed scheme has some limitations for the rotation attack. The highest and lowest NCC value observed between SI and RSI is 0.9815 and 0.9049 respectively. 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.

Analysing 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 correlation and SSIM of the recovered cover image as 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 on an average when 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 all of these attacks. A steganalyst can perform this identification in a variety of ways, but visual and statistical attacks are the most common. The efficacy of the built scheme is demonstrated in this paper using histogram attacks and RS analysis.

5.4.1 Histogram attack

To analyse 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 with respect to the amount of distortion induced by the embedding algorithm is evident. The obtained histograms does 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. To determine the noisiness of a group of pixels, a discrimination function is applied. 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

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	123.33	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	8.27	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

Table 11: Recovery evaluation of secret image in terms of PSNR and SSIM

Image	Embedded Secret Bits	Stego Image				
		R_M	R_{-M}	S_M	S_{-M}	RS value
Aeroplane	19832	25919	25717	13496	13765	0.0119
Baboon	29333	23864	23572	20260	20467	0.0113
Boat	22850	23968	23807	19328	19514	0.0080
Peepers	18805	22360	22334	18145	18221	0.0025

regardless of which flipping function is used, the number of R and S groups in a normal image would be the same. The RS value is derived using equation (15).

$$\frac{(|R_M - R_{-M}| + |S_M - S_{-M}|)}{(R_M + S_M)} \tag{15}$$

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 even after data embedding, the difference in a group of pixels is significantly less which is imperative for a data hiding scheme with high security.

6 Conclusion

In this paper, a robust scheme for data hiding is proposed 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, analysis 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. The suggested scheme could resist different image processing attacks as well, with a significant amount of recovery of the secret images. In the future, the objective will be to improve the payload capacity along with resistance to various geometrical attacks.

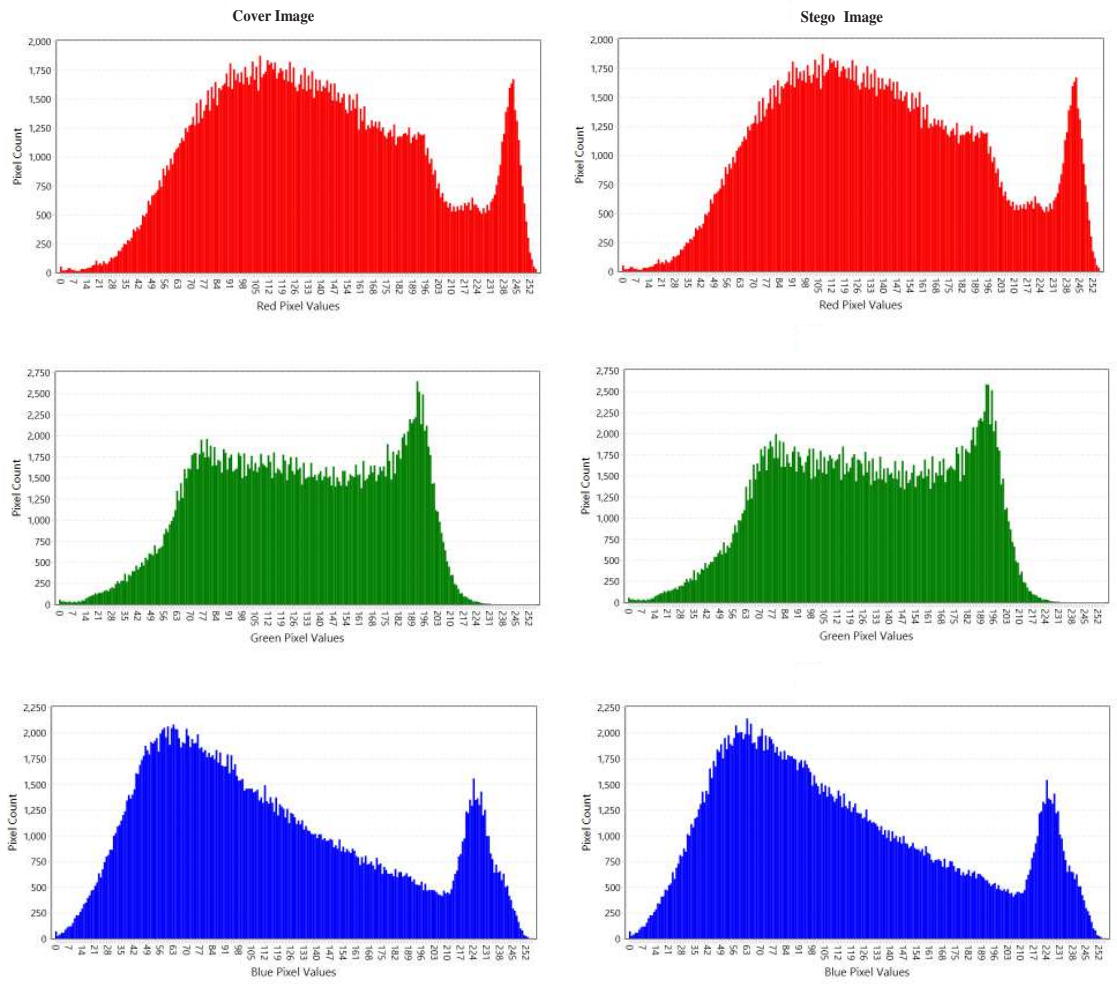

Fig. 14: Histogram analysis of color Cover and Stego image

Funding

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

Declaration of Competing Interest

We declare we have no competing interests.

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., IEEE. pp. 347–351.

Ashraf, Z., Roy, M.L., Muhuri, P.K., Lohani, Q.D., 2020. Interval type-2 fuzzy logic system based similarity evaluation for image steganography. Heliyon 6, 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. Multimedia tools and applications 76, 18451–18472.

Balasubramanian, C., Selvakumar, S., Geetha, S., 2014. High payload image steganography with reduced distortion using octonary pixel pairing scheme. Multimedia tools and applications 73, 2223–2245.

Chandramouli, R., Memon, N., 2001. Analysis of lsb based image steganography techniques, in: Proceedings 2001 International Conference on Image Processing (Cat. No. 01CH37205), IEEE. pp. 1019–1022.

Chang, C.C., Yu, Y.H., Hu, Y.C., 2008. Hiding secret data into an ambtc-compressed image using genetic algorithm, in: 2008 Second International Conference on Future Generation Communication and Network-

ing Symposia, IEEE. pp. 154–157.

Chowdhuri, P., Jana, B., Giri, D., 2018. Secured steganographic scheme for highly compressed color image using weighted matrix through dct. International Journal of Computers and Applications , 1–12.

Cox, I., Miller, M., Bloom, J., Fridrich, J., Kalker, T., 2007. Digital watermarking and steganography. Morgan kaufmann.

Dadgostar, H., Afsari, F., 2016. Image steganography based on interval-valued intuitionistic fuzzy edge detection and modified lsb. Journal of information security and applications 30, 94–104.

Demirci, R., 2006. Rule-based automatic segmentation of color images. AEU-International Journal of Electronics and Communications 60, 435–442.

Dubois, D.J., 1980. Fuzzy sets and systems: theory and applications. volume 144. Academic press.

Fridrich, J., Goljan, M., Du, R., 2001. Invertible authentication, in: Security and Watermarking of Multimedia contents III, International Society for Optics and Photonics. pp. 197–208.

Haibo, S.J.X.H.H., 2008. Discrete fourier transform-based information steganography. Journal of Huazhong University of Science and Technology (Nature Science Edition) , 08.

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.

Hamid, N., Yahya, A., Ahmad, R.B., Al-Qershi, O.M., 2012b. A comparison between using sift and surf for characteristic region based image steganography. International Journal of Computer Science Issues (IJCSI) 9, 110.

Jagadeesh, B., Kumar, P.R., Reddy, P.C., 2016. Robust digital image watermarking based on fuzzy inference system and back propagation neural networks using dct. Soft Computing 20, 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 Processing 163, 238–246.

Khashandarag, A.S., Navin, A.H., Mirnia, M.K., Mohammadi, H.H.A., 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, M.E., 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, C.F., Weng, C.Y., Kao, C.Y., 2019. Reversible data hiding using lagrange interpolation for prediction-error expansion embedding. Soft Computing 23, 9719–9731.

Lee, T.Y., Lin, S.D., 2008. Dual watermark for image tamper detection and recovery. Pattern recognition 41, 3497–3506.

Maity, S.P., Kundu, M.K., 2009. Genetic algorithms for optimality of data hiding in digital images. Soft computing 13, 361–373.

Molina-Garcia, J., Garcia-Salgado, B.P., 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 Processing: Image Communication 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 Computer Science 79, 912–921.

Niimi, M., Noda, H., Kawaguchi, E., 1999. Steganography based on region segmentation with a complexity measure. Systems and Computers in Japan 30, 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. IJ Network Security 19, 593–598.

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, D.R.I.M., 2020. Psnr vs ssim: imperceptibility quality assessment for image steganography. MULTIMEDIA TOOLS AND APPLICATIONS .

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

Sharma, S., Sharma, H., Sharma, J.B., 2021. Artificial bee colony based perceptually tuned blind color image watermarking in hybrid lwt-dct domain. Multimedia Tools and Applications , 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 Computing , 1–18.

UCID, 2017. Nottingham trent university, ucid image database. http://jasoncantarella.com/downloads/ucid.v2.tar.gz. 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, M.Y., Barani, M.J., Ayubi, P., 2020. A blind and robust color images watermarking method based on block transform and secured by modified 3-dimensional hénon map. Soft Computing 24, 771–794.

Wuerger, S.M., Maloney, L.T., Krauskopf, J., 1995. Proximity judgments in color space: tests of a euclidean color geometry. Vision research 35, 827–835.

Yuan, Z., Liu, D., Zhang, X., Wang, H., Su, Q., 2020. Dct-based color digital image blind watermarking method with variable steps. Multimedia Tools and Applications , 1–25.

Zadeh, L., 1965. Fuzzy sets. Information and Control 8, 338–353. URL: https://www.sciencedirect.com/science/article/pii/S001999586590241X, doi:https://doi.org/10.1016/S0019-9958(65)90241-X.

Zadeh, L.A., 1978. Fuzzy sets as a basis for a theory of possibility. Fuzzy sets and systems 1, 3–28.