Fuzzy Logic-Driven Variable LSB Image Steganography Using Gaussian Membership Functions

Vinita[1], Dr. Meenakshi[2], Navita[3]

[1,3]Research Scholar, Department of Mathematics, Maharshi Dayanand University, Rohtak

[2]Assistant Professor, Department of Mathematics, Maharshi Dayanand University, Rohtak

Abstract: ***With the growing transmission of multimedia data across open networks, there is a need for a method of communication in a secure and covert manner. Image steganography provides an effective approach to conceal secret information within digital images. This paper presents a LSB based image steganography scheme using fuzzy logic with Gaussian membership functions. Extraction of local image features, such as edge strength and texture is done using Sobel edge detection and local standard deviation filtering. A Sugeno-style fuzzy inference engine is employed to process the above-mentioned data so as to dynamically determine the bit rate (1-3) for each pixel. The proposed technique retains perceptual transparency for smooth areas and enhances the payload for the areas containing edges. Experimental results obtained for typical grayscale images demonstrate superior PSNR performance***

Keywords: Image steganography, Fuzzy logic, Gaussian membership function, Variable LSB, Sobel operator, Texture analysis

1. Introduction

The fast development of digital communications and multimedia data transmission over open networks has made the protection of the security and confidentiality of sensitive information a major challenge. However, traditional cryptography techniques protect the content of a message but not its existence, and thus encrypted data is open to suspicion and potential attacks. Steganography deals with this problem by hiding secret information in digital media that appears innocent, such as text, images, audio or video. Among other types of carriers, digital images are one of the most popular steganographic carriers due to its high redundancy, ease of transmission and high availability on the internet. Edge and texture regions are less prone to modification and can thus store more data without perceptual degradation. Such local features can be exploited to improve both imperceptibility and payload capacity [1]. One of the most popular steganography techniques is Least Significant Bit (LSB) substitution, which is a simple technique with a high embedding capacity. However, fixed-LSB schemes embed data uniformly in the image, which causes visible distortion [2]. The classical LSB methods are simple and computationally efficient, but they embed data uniformly over an image, which often leads to perceptible distortion and makes them vulnerable to statistical detection [3]. Fuzzy logic systems [4] have been extensively investigated in adaptive steganography because of their capability to manage uncertainty and model imprecise information. Unlike crisp threshold-based methods, fuzzy systems allow smooth transitions between embedding levels. Gaussian membership functions provide continuity and robustness, and Sugeno-type fuzzy inference systems offer computational efficiency. Zhang et al. [5] utilized texture masking for adaptive allocation of embedding capacity, which showed increased payload and decreased detectability. Fuzzy logic systems have become one of the efficient ways of coping with the uncertainty inherent in adaptive steganography. While hard-threshold algorithms have certain limitations, fuzzy logic systems allow for gradual change from one embedding level to another on the basis of vague inputs like the edge magnitude or texture complexity. Wu and Tsai [6] suggested a method of adaptive LSB algorithm, where the number of bits that can be embedded in each pixel varies depending on edge intensity and is more imperceptible than fixed-LSB methods [5]. Rajput et al. [7] presented fuzzy-based algorithm for steganography, where embedding rate is controlled by fuzzy rules considering edge intensity and local variance. Hossain and Rahman [9] presented an adaptive LSB technique utilising a Sugeno fuzzy logic that varies the number of bits per pixel according to the local intensity change and edge strength, resulting in higher PSNR and payload capacity. Unlike Mamdani systems, Sugeno systems [9] yield crisp output which suits well the bit-level LSB steganography. Vinita et al. [10] present a new technique for grayscale image steganography called mshEdgeGrayFT1 which incorporates a Mamdani fuzzy inference system with the LSB approach, where edge identification is based on pixel intensity differences after Gaussian filtering. It is proven by experiments conducted using standard performance measures that embedding secret information into edge pixels selected through fuzzy approach makes mshEdgeGrayFT1 superior to other techniques. Motivated by these advantages, this paper presents a fuzzy-driven variable LSB steganography scheme based on Gaussian membership functions.

2. Feature Extraction

Edge strength characterizes local intensity discontinuities and is computed using the Sobel gradient operator. Let $I(x,y)$ denote the grayscale image. The horizontal and vertical Sobel kernels are defined as [3]:

$$S_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix},\ S_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix}$$

The corresponding gradient components are obtained by convolution:

$$G_x(x,y) = I(x,y) * S_x,$$
$$G_y(x,y) = I(x,y) * S_y$$

The edge strength at the pixel location (x, y) is computed as the gradient magnitude:

$$E(x,y) = \sqrt{G_x^2(x,y) + G_y^2(x,y)} \qquad (1)$$

The resulting edge magnitude map is normalized to the range [0,1] to ensure scale invariance:

$$E_n(x,y) = \frac{E(x,y)}{E_{\max}} \qquad (2)$$

where $E_{\max}$denotes the maximum possible gradient magnitude for an 8-bit image.

2.2 Texture Variation Using Local Standard Deviation

Texture variation is quantified using the local standard deviation computed over a 3 × 3 sliding window centred at pixel (x,y):

$$T(x,y) = \sigma_{3\times3}\big(I(x,y)\big) \qquad (3)$$

Higher standard deviation values correspond to highly textured regions, which can tolerate higher embedding rates. The texture feature is normalised to the range [0,1] as:

$$T_n(x,y) = \frac{T(x,y)}{T_{\max}} \qquad (4)$$

where $T_{\max}$represents the maximum possible standard deviation for an 8-bit grayscale image.

2.3 Gaussian Membership Functions

Gaussian membership functions are employed due to their smoothness and robustness to noise. The general form of a Gaussian membership function is given by [4]:

$$\mu(x; c, \sigma) = \exp\left(-\frac{(x-c)^2}{2\sigma^2}\right) \qquad (5)$$

where x denotes the input feature value, c is the centre of the membership function, and σ controls its spread.

Each input variable **edge strength** and **texture variation** is fuzzified into three linguistic terms:

- Low
- Medium
- High

3. Literature Survey

Over the last decade, the field of image steganography has witnessed significant advancements, particularly in techniques that aim to balance payload capacity, imperceptibility**,** and robustness against attacks. Traditional LSB-based methods, while simple and computationally efficient, embed data uniformly across an image, often leading to noticeable distortion and susceptibility to statistical detection [5,6]. To overcome these limitations, researchers have explored adaptive and intelligent embedding schemes that consider local image characteristics. It has been well established that human visual sensitivity is lower in edge-rich or highly textured regions of an image. Techniques that exploit edge and texture information can therefore embed more data in these regions without perceptible distortion. Wu and Tsai (2003) proposed an edge-adaptive LSB method in which the number of bits embedded per pixel depends on local edge strength, showing improved imperceptibility compared to fixed-LSB schemes [7]. Similarly, Zhang et al. (2011) used texture masking to adaptively allocate embedding capacity, demonstrating enhanced payload and reduced detectability [8]. Fuzzy logic systems have emerged as an effective approach to handle the inherent uncertainty in adaptive steganography. Unlike hard-threshold methods, fuzzy systems allow smooth transitions between embedding levels based on imprecise inputs such as edge magnitude or texture complexity. Rajput et al. (2016) proposed a fuzzy-driven steganographic approach in which the embedding rate is determined by fuzzy rules that consider both edge strength and local variance, thereby improving imperceptibility [9]. Gaussian membership functions have been widely employed to provide smooth, continuous mapping from input features to output embedding capacity [10]. Unlike Mamdani systems, Sugeno systems produce crisp outputs that align well with the discrete nature of bit-level LSB steganography. Hossain and Rahman (2018) demonstrated a Sugeno-based adaptive LSB method that adjusts the number of bits per pixel based on local intensity variation and edge magnitude, achieving higher PSNR and payload capacity [11]. Vinita et al. (2025) propose a novel grayscale image steganography algorithm, **mshEdgeGrayFT1**, integrating a Mamdani fuzzy inference system with the LSB technique, where pixel intensity differences guide edge detection after Gaussian filtering. Experimental results using standard evaluation metrics show that embedding confidential data in fuzzy-identified edge pixels enables **mshEdgeGrayFT1** to outperform existing methods [12].

4. Proposed Work

This section describes the proposed fuzzy-driven variable LSB image steganography method for grayscale images. The technique adaptively determines the number of bits embedded per pixel using a Sugeno-type fuzzy inference system (FIS). Unlike conventional edge-based schemes, the proposed method jointly exploits edge strength and texture variation, enabling improved imperceptibility and embedding capacity.

The complete framework consists of two phases: data embedding and data extraction, both governed by identical feature extraction and fuzzy decision mechanisms.

4.1 Embedding and Extraction Algorithm

Embedding Phase

1) Convert the cover image to grayscale and normalize pixel intensities.
2) Compute local edge strength and texture variation.
3) Fuzzify the extracted features using Gaussian membership functions.
4) Apply a Sugeno-type fuzzy inference system to generate a Bits-Per-Pixel (BPP) map.

Paper ID: SR26914170007 DOI: https://dx.doi.org/10.21275/SR26914170007

5) Embed secret bits into each pixel according to the adaptive LSB level.

Extraction Phase
1) Recompute edge strength and texture features from the stego image.
2) Apply the same fuzzy inference system to regenerate the BPP map.
3) Extract the embedded bits and reconstruct the secret message.

4.2 Cover Image Representation

Let C denote the selected cover image. If C is an RGB image, it is converted into a grayscale image G of size $W \times H$, where each pixel intensity $P_{ij} \in [0,255]$.

For illustration, a sample 3×3 grayscale block is considered:

$$G = \begin{bmatrix} 12 & 34 & 45 \\ 55 & 67 & 89 \\ 200 & 210 & 220 \end{bmatrix}$$

4.3 Feature Extraction

4.3.1 Edge Strength Computation
Edge strength is calculated using Sobel gradient operators in horizontal and vertical directions. For a central pixel $p_{22} = 67$, first we calculated G_x and G_y by using the horizontal and vertical Sobel kernels. After this, the gradient magnitude is computed by using (1). $Here,$

$$G_x = 121, G_y = -715$$
$$E = \sqrt{525866} \approx 725$$

For an 8-bit image, the maximum Sobel gradient magnitude is approximately $E_{\max} = 1020$. The normalised edge strength is computed by using (2):

$$E_n = \frac{E}{E_{\max}} = \frac{725}{1020} \approx 0.71$$

4.3.2 Texture Variation Computation
Texture complexity is quantified using the local standard deviation computed over a 3×3window:

$$T(x,y) = \sigma_{3\times3}(I(x,y))$$

For the sample block, the mean intensity is:

$$\mu \approx 103.56$$

and the standard deviation is:

$$\sigma \approx 77.8$$

The texture feature is normalised as:

$$T_n = \frac{\sigma}{128} = \frac{77.8}{128} \approx 0.61$$

4.4 Fuzzification Using Gaussian Membership Functions

To model gradual transitions in natural images, both features are fuzzified using Gaussian membership functions. Each input variable is represented by three linguistic terms: **Low**, **Medium**, and **High**. The parameters of these membership functions are summarised in Table X, where c denotes the center and σ represents the standard deviation.

Gaussian Membership Function Parameters

Edge Strength Membership Functions

Fuzzy Set	Center c	Standard Deviation σ
Low	0.15	0.10
Medium	0.30	0.18
High	0.58	0.22

Texture Membership Functions

Fuzzy Set	Center c	Standard Deviation σ
Low	0.15	0.12
Medium	0.50	0.18
High	0.90	0.22

Take T=0.6 and E=0.7 as input variables and compute the fuzzy membership values for low, medium and high fuzzy sets associated with these inputs using (5).

$$\mu_{\text{Low}}(0.6) = 0.0009,\ \mu_{\text{Medium}}(0.6) = 0.857, \mu_{\text{High}}(0.6) = 0.395$$

$$\mu_{\text{Low}}(0.7) = 2.7 \times 10^{-7}, \mu_{\text{Medium}}(0.7) = 0.085, \mu_{\text{High}}(0.7) = 0.861$$

4.5 Sugeno Fuzzy Inference System

A Sugeno-type FIS with two inputs (edge strength and texture variation) and one output (bits per pixel) is employed.

4.5.1 Fuzzy Rule Base

Rule 1: If Edge is High OR Texture is High **THEN** Output (Embedding Strength / Bits per Pixel) is **3.**
Rule 2: If Edge is Medium OR Texture is Medium **THEN** Output (Embedding Strength / Bits per Pixel) is 2.
Rule 3: If Edge is Low OR Texture is Low **THEN** Output (Embedding Strength / Bits per Pixel) is 1.

4.6 Rule Evaluation

The implication is implemented for the three fuzzy rules, and the intermediate results are inferred for each rule as given below:

Using the max operator for OR conditions, the firing strengths are computed as:

$$w_1 = \max(\mu_{High}\ (0.7), \mu_{High}(0.6)) = \max(0.861, 0.395) = 0.861$$
$$w_2 = \max(\mu_{Medium}(0.7), \mu_{Medium}\ (0.6)\ ) = \max(0.085, 0.857) = 0.857$$
$$w_3 = \max(\mu_{Low}\ (0.7), \mu_{Low}(0.6)) = \max(0, 0.0009) = 0.0009$$

4.7 Defuzzification

Sugeno weighted average defuzzification is applied to obtain the final embedding capacity:

$$\text{BPP} = \frac{3w_1 + 2w_2 + 1w_3}{w_1 + w_2 + w_3}$$
$$\text{BPP} = \frac{(3 \times 0.861) + (2 \times 0.857) + (1 \times 0.0009)}{0.861 + 0.857 + 0.0009}$$

$$\text{BPP} = \frac{4.2979}{1.718} \approx 2.5$$

Thus, **2 least significant bits** are used for embedding.

4.8 Let the secret text in binary form be 01001001 and we want to embed 2 bits in the identified pixel i.e. 67. The binary representation of 67 is $(01000011)_2$. Now we replace the first two secret bits **("01")** by embedding them in the two LSBs of the pixel. i.e. $(010000\underline{11})_2 \rightarrow (010000\underline{01})_2$. Thus, the pixel value changes from **67 to 65**, producing negligible visual distortion.

Extraction: The same process is repeated during the extraction of the secret message from hidden pixels.

5. Dataset and Quality Metrics

The performance of the proposed fuzzy-driven variable LSB steganography algorithm is evaluated using five widely adopted benchmark grayscale images: Lena, Baboon, Jet, Barbara, and Sailboat, each of size 256 × 256 pixels, as shown in Fig. 1(a)–(e). These images represent diverse visual characteristics, including smooth regions, strong edges, and highly textured areas.

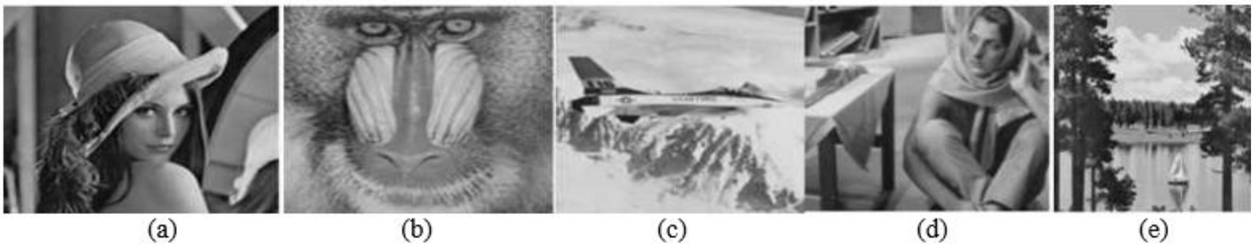

Figure 1: Cover images

To assess the imperceptibility and embedding efficiency of the proposed method, the following quality metrics are employed:

1) Peak Signal to Noise Ratio (PSNR): This formula is used to calculate the difference between the cover image and stego image. The PSNR is defined as:

$$\text{PSNR}= 10.\log_{10}\frac{255^2}{MSE} \text{ (dB)}$$

Where the MSE is the mean square error between the pixels in cover image and stego image. For a cover image with height H and width W, MSE is defined as:

$$\text{MSE}= \frac{\sum_{i=1}^{H}\sum_{j=1}^{W}(p_{ij} - p'_{ij})2}{(\text{H} * \text{W})}$$

Where p_{ij} and p'_{ij} refer to pixel values of the cover image and the stego images. Obviously, a higher PSNR shows that stego image is very close to the cover image.

(2) Payload Capacity (bits per pixel, bpp)= Payload capacity refers to the maximum amount of secret data that can be embedded into a cover image without causing noticeable degradation in image quality. It is usually measured in bits per pixel (bpp) or the total number of embedded bits.

$$\text{Payload Capacity (bpp)} = \frac{\text{Total Embedded Bits}}{\text{Total Number of Pixels}}$$

6. Experimental Simulations and Performance Analysis

This work proposes a fuzzy-logic–driven variable LSB image steganography scheme that adaptively determines embedding capacity using Gaussian membership functions. Unlike conventional fixed-LSB techniques, the proposed approach exploits local edge strength (computed via Sobel gradient operators) and texture variation (measured using local standard deviation) to guide the embedding process.

A Sugeno-type fuzzy inference system processes these features to dynamically assign **1–3 bits per pixel**, thereby achieving an effective trade-off between imperceptibility and payload capacity. The secret message is converted to binary and embedded in the grayscale cover image based on the fuzzy decision map. During extraction, identical feature-computation and fuzzy-inference procedures are applied to the stego image to accurately retrieve the hidden message.

Table 1: Results of Image quality metrics

Image	PSNR	Payload
Lena	39.57	2.39
	35.26	3.39
Baboon	40.35	2.12
	41.49	1.98
Jet	38.78	2.48
	37.64	2.78
Barbara	41.14	2.19
	40.58	2.05
Sailbaot	38.18	2.71
	37.59	2.81

Table 2: Comparative overview of the previous steganographic schemes with proposed work

Transversing method	Payload	PSNR (dB)
Chen-et al.	2.1	37.5
	2.73	32
Jung-et al.	2.35	31.94
Lee-et al	2.14	40.39
Tseng-et al.	2.41	38.18
	3.16	33.58
S. Dhargupta et al.	2.39	37.02
	3.39	28.16
Proposed work	2.39	39.57
	3.39	35.26

7. Conclusion

A fuzzy logic–driven variable Least Significant Bit (LSB) image steganography technique based on Gaussian membership functions was presented in this work. Local image features, including edge strength and texture variation, were extracted using Sobel edge detection and local standard deviation filtering. A Sugeno-type fuzzy inference system was utilized to adaptively select the number of embedded bits per pixel. This adaptive embedding strategy increased payload capacity in edge regions while preserving imperceptibility in smooth areas. Experimental results on standard grayscale images demonstrated high PSNR values and improved visual quality. The proposed method achieved superior payload capacity compared to fixed-LSB and existing adaptive schemes. Overall, the results confirm the effectiveness and robustness of the proposed approach for secure image steganography.

References

[1] Sumathi, C. P., Santanam, T., & Umamaheswari, G. (2014). A study of various steganographic techniques used for information hiding. *arXiv preprint arXiv:1401.5561*.

[2] Thien, C. C., & Lin, J. C. (2003). A simple and high-hiding capacity method for hiding digit-by-digit data in images based on modulus function. *Pattern recognition*, *36*(12), 2875-2881.

[3] Chen, W. J., Chang, C. C., & Le, T. H. N. (2010). High payload steganography mechanism using a hybrid edge detector. *Expert Systems with applications*, *37*(4), 3292-3301.

[4] Dhargupta, S., Chakraborty, A., Ghosal, S. K., Saha, S., & Sarkar, R. (2019). Fuzzy edge detection-based steganography using modified Gaussian distribution. *Multimedia Tools and Applications*, *78*(13), 17589-17606.

[5] Wong, K., Yang, R. K. and Hui, & J. C. D. , (2003). A new method for image steganography based on LSB substitution. *IEEE Transactions on Image Processing*, 12(12), 1580–1592.

[6] Wu, M., & Tsai, W. (2003). Adaptive data hiding in images with improved imperceptibility. *Pattern Recognition Letters, 24*, 1615–1625. https://doi.org/10.1016/S0167-8655(02)00406-1.

[7] Zhang, J., Yang, X., & Li, Q. (2011). Texture-adaptive LSB image steganography. *Journal of Visual Communication and Image Representation, 22*, 178–186. https://doi.org/10.1016/j.jvcir.2010.12.002

[8] Rajput, P., Singh, S., & Sharma, R. K. (2016). Fuzzy logic based adaptive image steganography. *International Journal of Computer Applications, 141*(4), 1–6.

[9] Hossain, M., & Rahman, M. (2018). Sugeno-type fuzzy inference system for adaptive LSB image steganography. *Multimedia Tools and Applications, 77*, 25011–25028. https://doi.org/10.1007/s11042-018-6034-6

[10] Zadeh, L. A. (1965). Fuzzy sets. *Information and Control, 8*, 338–353. https://doi.org/10.1016/S0019-9958(65)90241-X

[11] Hossain, M. S., & Rahman, M. (2018). Sugeno-based adaptive LSB steganography for digital images. *Journal of Visual Communication and Image Representation, 54*, 45–55. https://doi.org/10.1016/j.jvcir.2018.03.012

[12] Yadav, V., Hooda, M., & Dhaka, N. (2025). A novel steganography algorithm based on interval Type-1 fuzzy logic system. *Statistics, Optimization & Information Computing, 14*(3), 1512–1524.