

A Novel Payload Image Steganography Using Type-2 Fuzzy Logic and Edge Detection

Hala Salih Yusuf ^[1], Hani Hagra ^[2]

^[1] Computer Science, Sudan University of Science and Technology College of Graduate Studies, Khartoum - Sudan

^[2] The Computational Intelligence Centre, University of Essex School of Computer Science and Electronic Engineering - U.K

ABSTRACT

Steganography is an art and a science that includes communicating the secret message in an appropriate multimedia carrier, e.g., image, audio and video files. In the last decade, image steganography has received a lot of attention due to the lowering of the cost of storage media, which has allowed for a wide use of a large number of images.

This paper presents an image steganography method based on the combination of Least Significant Bit (LSB) substitution and type-2 fuzzy logic edge detection due to their ability to handle the high level of uncertainty present in images. Based on this, the type-2 fuzzy system will be employed for detection of more of edge pixels which exploit in hiding a large amount of data than non-edge pixels. Many experiments were conducted to measure the performance of the proposed method. When we compared our results with the previous work, the results showed that our proposed system provides better stego image quality, as well as higher embedding capacity than previous work. We used metrics, like Peak Signal to Noise Ratio (PSNR) and human visual system (HVS), to measure the quality of stego image. The BSD300 dataset color images were used in our experiments.

Keywords: - Image Steganography, LSB, type-1 fuzzy logic, type-2 fuzzy logic, edge detection, PSNR.

I. INTRODUCTION

In the digital world, Information security is very important for transmitting data in a communication network. There are two important methods for providing security to information which are cryptography and steganography [1]. Steganography is a Greek term means "covered writing", which an art and science that traces back to ancient times. Steganography is hiding secret message in ways that prevent the detection of secret messages. It difference from cryptography that encrypting the secret message into un-understood form [1][2][3].

Steganography can be classified into four types: text steganography, image steganography, audio/video steganography and protocol steganography [4]. During the last decade, Image steganography where a secret message embedding within an image, has been widely used due to weaknesses of the human visual system (HVS) also the lowering of the cost of image storage and communication [5].

Image steganography techniques can be divided into two categories: spatial domain technique and frequency domain technique. In the first technique, the secret message is embedded directly in the intensity of the pixels, while in the second technique, images are first transformed to frequency domain and then, the secret message is embedded in the transformed image frequency domain [2][5].

There are interrelationships between the three requirements of steganography: capacity, robustness, and imperceptibility, as shown in Figure 1. Capacity indicates the amount of secret information that can be embedded within the cover image without damaging the integrity of the cover image. Robustness refers to

the quantity of modification that the stego image can resist without an attacker can detect hidden information using image processing operations. The concept of imperceptibility is the most important requirement of the steganography system, as the strength of the steganography system relies on its ability to be unnoticed by the human visual system (HVS). It is an interesting issue to balance the trade-off of these three requirements in the fields of information hiding [3][6].

The security of any steganography technique relies on the selection of pixels for embedding. Pixels in textured and noisy areas are better choices for embedding since they are hard to model. Pixels in the edge area can be seen as noisy pixels, their intensities are either lower or higher than their neighbor pixels, because of a sudden change in the coefficient gradient. Due to these sharp changes in the visual and statistical properties, edges are hard to model in comparison with pixels of the smoother area [3].

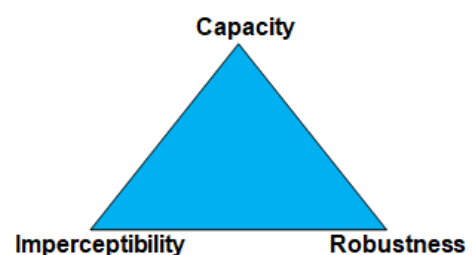


Fig. 1 Magic triangle model of Steganography.

II. LEAST SIGNIFICANT BIT (LSB)

The most obvious method to conceal secret information within an image is the least significant bit (LSB) technique, which provides great performance in terms of high payload and low computational complexity. In this technique, secret information is embedded in the least significant bits of pixels in the image, which means that the value of the least bit of pixels in the cover image is utilized to indicate the secret message [5] [6] [7].

The rest of the paper is organized as follows: the following section presents related work, this will be followed by describing the proposed method, then section presents the experimental results. Finally, presenting the paper conclusions.

III. RELATED WORK

In [7] J. Bai et al. presented a scheme based on the LSB substitution method, combined with the edge detector, which uses the principle that edge areas can tolerate a larger number of embedded bits more than smooth areas according to HVS. In their scheme, they generate Most-Significant-Bit (MSB) image from the cover image by clearing the last 5 LSBs of each pixel in the original image for edge detection. They employed the 5 LSBs for embedding the secret data while 3 MSBs of all pixels remain unchanged. They classify the pixels of the cover image into two types, which are non-edge pixels and edge pixels, respectively. Every cover pixel in the first type contains 'x' secret message bits, and the second type contains 'y' secret message bits, using LSBs substitution. For these two types, pixels are embedded by the k-LSB substitution, where the value k equals either x or y, which is decided by the edge information. The secret key K is shared between the sender side and the receiver side. For example, suppose the block is = 4 pixels, that is, P1 = [10011011], P2 = [01111110], P3 = [01011000], P4 = [10011100], x = 2, y = 4, then the secret bit S = '101001111110'. Based on the edge information of these four pixels, we know that P1 and P4 are edge pixels and P2 and P3 are non-edge pixels. P1 and P4 pixels will include 4 bits of a secret message, while P2 and P3 will include 2 bits of a secret message. These four pixels will switch to P1' = [1001 1010], p2' = [01111101], P3' = [010110 11] and P4' = [1001 1110]. In the extraction phase, the receiver retrieves the two parameters x and y from the last four pixels of the image. And also, the edge information is determined the same as in the embedding phase.

In [6] Hala et al. proposed a high payload LSB steganography method, which developed into two sides, the sender's side that treats with the embedding process, and the receiver's side that treats with the extraction process. In the embedding process, they used edge images obtained from gradient type-1 fuzzy logic edge detection combined with canny edge detector to increase edge pixels. They used RGB color image as the cover image for hiding the secret message. They utilize two parameters x and y. If the pixel is an edge pixel the number of secret bits to be embedded will be x bits and if the pixel is a non-edge pixel the number of secret bits to be embedded will be y bits. The first bit in red color will be (0 or 1) to indicate is this (non-edge or edge) pixel respectively. Pixels are embedded by the M-LSB substitution, where the value of M equals either x=9 or y=3 depends on the edge information. Let's assume that 4 pixels [P1, P2, P3, and P4],

are read from the cover image I. According to the edge information of these four pixels, we know that both the first and the second pixels are edge pixels and that the third and the fourth pixels are non-edge pixels. Consider that the secret bit-streams s='110010100100111010011110'. The 9 LSBs of the first and the second pixels are replaced with the corresponding secret bits, and the first bit of red will be 1 (because it is an edge pixel). Similarly, the 3 secret bits are embedded into the third and fourth pixels by the LSB substitution method, and the first bit of red will be 0 because it is a non-edge pixel. The values of the pixels before and after the embedding operation will be demonstrated in Figure 2, where the green color indicate the edge information and the red color indicate embedded data.

In the extraction process, the receiver extracts the first bit in red color if it's 0 or 1 to determine the parameter value of m. If the first bit of red color is 0, this means the value of m will be 3 bits (1bit in red, 1bit in green and 1bit in blue color.), and if it's 1, this means the value of m will be 9 bits (3bits in red, 3bits in green and 3bits in blue color). Thus, the secret data will be accurately extracted.

```

P1 (11011100 11101011 11101110)
P2 (11010110 11101011 11101100)
P3 (10100001 10000001 01110100)
P4 (10100101 01111011 01100011)

P1 (11011101 11101010 11101100)
P2 (11011001 11101111 11101010)
P3 (10100000 10000001 01110101)
P4 (10100110 01111011 01100010)
    
```

Fig 2: Example of the proposed embedding operation

In [8] I et al. proposed a new general type-2 fuzzy logic method for edge detection applied to color format images. The proposed algorithm combines the methodology based on the image gradients and general type-2 fuzzy logic theory to provide a powerful edge detection method. General type-2 fuzzy inference systems are approximated using the α -planes approach. The edge detection method is tested on a database of color images with the idea of illustrating the advantage of applying the fuzzy edge detection approach on color images against grayscale format images, and also when the images are corrupted by noise. They compare the proposed method based on general type-2 fuzzy logic with other edge detection algorithms, such as ones based on type-1 and interval type-2 fuzzy systems. Simulation results show that edge detection based on a general type-2 fuzzy system outperforms the other methods because of its ability to handle the intrinsic uncertainty in this problem.

IV. TYPE-1 FUZZY LOGIC SYSTEM

Fuzzy Logic was introduced by Lofti A. Zadeh in 1965, He is considered as the founding father of the fuzzy logic field [6][9][10]. Fuzzy Logic attempt to mimic the way of the human decision-making methods by means of the use of fuzzy set [11] [12][13]. It treats with uncertainty and vague information associated with the FLS inputs and outputs by fuzzy sets as shown in Figure 3, where a fuzzy set is characterized by a Membership

Function (MF), i.e. the membership grade for each element is a crisp number in the range from 0 to 1. Here 0.0 represents absolutely false and 1.0 represents absolutely true [14][15][16][17]. Fuzzy Logic Systems (FLSs) provide a good decision response when facing uncertainties, due to the smooth transition between the fuzzy sets. FLSs employ a number of if-then rules, which are easy to analyze and understand by the ordinary user [10].

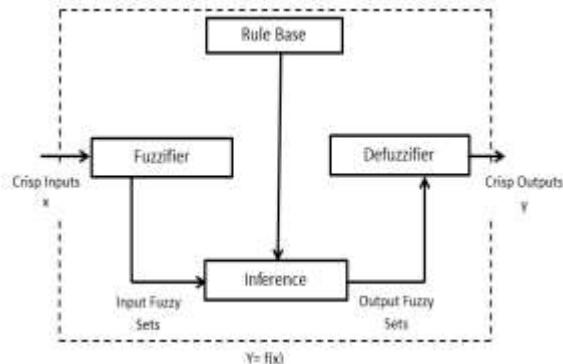


Fig. 3 Fuzzy Logic System [18].

Fuzzy Logic Systems (FLSs) provide a good decision response when facing uncertainties, due to the smooth transition between the fuzzy sets. FLS consists of four components which are fuzzifier, inference engine, rules, and defuzzifier. The fuzzifier maps a crisp input to fuzzy sets and this mapping can be expressed as $y = f(x)$ [6][13]. Rule Base can be obtained using data from the system, or designed by experts or consultants. The FLS inference engine employs the Rule Base to transform fuzzy sets into other fuzzy sets, and control the way of combining the rules. The Defuzzifier converts the fuzzy sets obtained by the inference engine to crisp input [13]. Type-1 fuzzy sets handle the uncertainties related to the FLS antecedents and Consequents by using precise and crisp membership functions [19][20][21]. When the type-1 membership functions have been selected, all the uncertainty disappears, due to type-1 membership functions are completely precise [14].

V. TYPE-2 FUZZY LOGIC SYSTEM

The type-2 fuzzy logic system shown in Figure 4 was introduced by Lotfi Zadeh in 1975 [22][23][24][25]. The type-2 fuzzy sets are an extension of ordinary type-1 fuzzy sets where a type-2 fuzzy set is able to fully handle the high levels of uncertainties associated with control applications. A type-2 fuzzy set whose membership function (MF) grades themselves are typ-1 fuzzy sets. A type-2 membership grade can be any subset in [0, 1] which called the primary membership; and corresponding to each primary membership, there is a secondary membership grade is a crisp number in [0, 1] [23][24][26]. As shown in Figure5, the type-2 fuzzy set has a three-dimensional membership function and includes a footprint of uncertainty (FOU). A FOU can be described in terms of an upper membership function and a lower membership function; that gives additional degrees of freedom to make it reasonable to handle a high level of uncertainties [27][28][29][30][31]. The type-2 fuzzy sets are beneficial where it is difficult to determine the exact and precise membership

functions [25] [31].

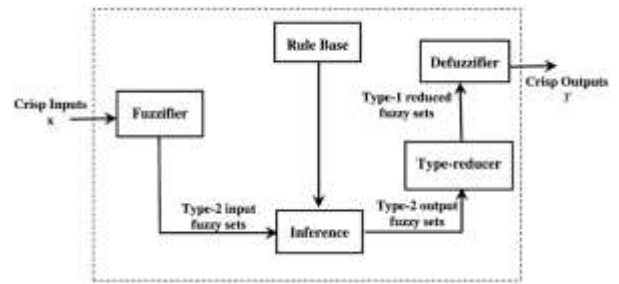


Fig. 4 Type 2 Fuzzy Logic System [14].

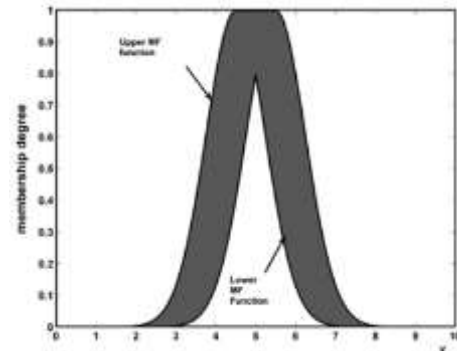


Fig. 5 Interval Type 2 Fuzzy Set.

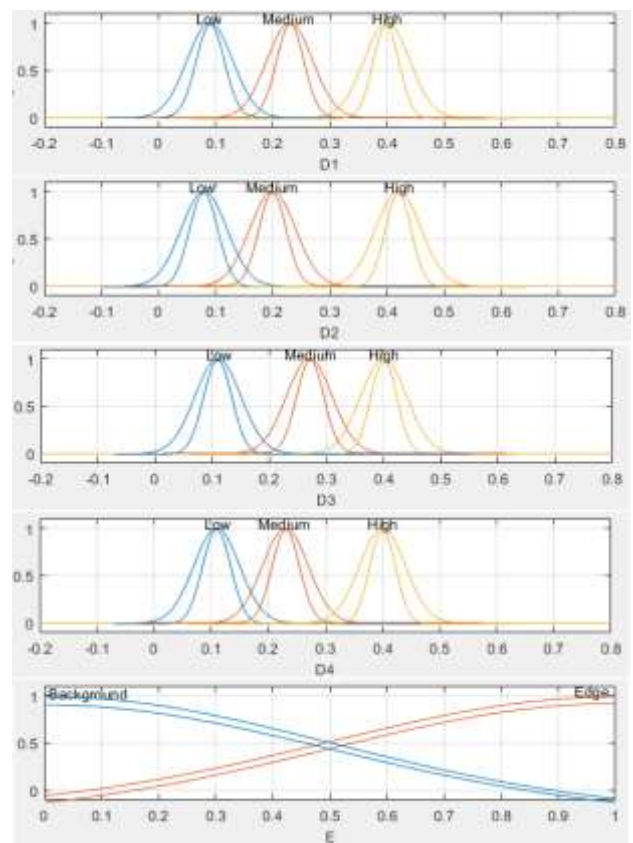


Fig. 6 T2FIS membership function for the inputs (D1, D2, D3 and D4) and the output E.

We can represent the type-2 fuzzy set (T2 FS) \tilde{A} , by the membership function:

$$\tilde{A} = \{(x,u), \mu_{\tilde{A}}(x,u) \mid \forall x \in X, \forall u \in J_x \subseteq [0,1]\} \quad (1)$$

where $0 \leq \mu_{\tilde{A}}(x,u) \leq 1$. Another expression for \tilde{A} is, When \tilde{A} is continuous it commonly expressed as:

$$\tilde{A} = \int_{x \in X} \int_{u \in J_x} \mu_{\tilde{A}}(x,u)/(x,u) J_x \subseteq [0,1] \quad (2)$$

The integral sign denote the union over all admissible x and u.

When \tilde{A} is discrete it commonly expressed as:

$$\tilde{A} = \sum_{x \in X} \sum_{u \in J_x} \mu_{\tilde{A}}(x,u)/(x,u) \quad (3)$$

The summation sign denote the collection over all admissible x and u.

The primary membership of x represents by $J_x \subseteq [0,1]$, and $\mu_{\tilde{A}}(x,u)$ is a type-1 fuzzy set also known as the secondary set. Thus, a type-2 membership grade can be any subset in [0, 1], the primary membership, and corresponding to each primary membership, there is a secondary membership (which can also be in [0, 1]) that defines the possibilities for the primary membership.

VI. EDGE DETECTION

Edge detection is an important area of digital image processing and machine vision due to that edges are considered to be the significant features for analyzing and extracting the most essential information contained in images. Basically, edge detection is adopted to indicate the abrupt changes in the intensity of an image and identify the current pixel a non-edge pixel or an edge pixel [32][33]. There are many classical or standard edge detection such as Sobel, Prewitt, Laplacian and Canny operators are already available in the literature [7][10][34][35]. In recent years, many fuzzy techniques have been presents to edge detection. In this paper, the proposed method was developed based on Fuzzy edge detectors are based on type-2 fuzzy logic.

A. Gradient Edge Detector

There are some methods to carry out the edge detection process; most of them are based on the image gradient approach, which is obtained with the first derivative of an image. In this paper, the edge detection method is performed by calculating the image gradients with the Euclidean distance; which is the most important and the most used method. This process contains calculating four image gradients to indicate the edge direction based on a 3x3 matrix (D_i , for $i=1 \dots 4$) (Figure 7 illustrates that). Each matrix position (D_i), of figure7, is represented in Figure 8, where f indicates the image x-axis the rows, and y-axis the columns [8].

According to these positions, the Euclidean distance is applied to calculate the gradients D_i using the eq (4). Gradient magnitude, or The edges E, can be calculated with the Equation (5) [6][8][33].

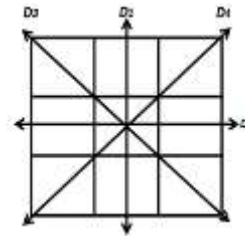


Fig. 7 3 x 3 Matrix indicating direction of the four gradients D_i .

$P_2 = f(x-1, y-1)$	$P_3 = f(x-1, y)$	$P_4 = f(x-1, y+1)$
$P_1 = f(x, y-1)$	$P_5 = f(x, y)$	$P_6 = f(x, y+1)$
$P_7 = f(x+1, y-1)$	$P_8 = f(x+1, y)$	$P_9 = f(x+1, y+1)$

Fig. 8 Matrix position.

$$D_1 = \sqrt{(p_5 - p_2)^2 + (p_5 - p_8)^2} \quad (4)$$

$$D_2 = \sqrt{(p_5 - p_4)^2 + (p_5 - p_6)^2}$$

$$D_3 = \sqrt{(p_5 - p_1)^2 + (p_5 - p_9)^2}$$

$$D_4 = \sqrt{(p_5 - p_3)^2 + (p_5 - p_7)^2}$$

$$Edge = D_1 + D_2 + D_3 + D_4 \quad (5)$$

B. Edge Detection Process Based On the Gradient Approach and Type-2 Fuzzy Logic System

This section presents The fuzzy logic methodology for edge detection, using gradient magnitude consists of using Eq (4) to obtain the gradients in the four directions (D_1, D_2, D_3, D_4) and employ them as inputs to a fuzzy inference system (FIS), instead of Eq. (5).

In this paper, the T2FIS has been used a singleton Mamdani type, which was designed to contain four inputs (D_1, D_2, D_3 , and D_4), and only one output. The inputs and outputs are fuzzified using Gaussian membership functions with uncertain mean; each input has three linguistic values: low, medium, and high to determine the grade to which the evaluated gradient corresponds, to be the output edge. Each output has two linguistic values: edge and background to produce the gradient magnitude edge detection. The T2FIS Gaussian membership function is illustrated in Figure 4.

The parameters required for the T2FIS MFs are expressed in Equation (6), and these are calculated depending on the image gradient values, i.e., considering the image of Figure 9, the parameters are obtained with Equations (7) to (11)[6][8] [33].

$$\mu(x,u) = \text{igausmtype2}(x, [\sigma, m1, m2]) \quad (6)$$

$$low = \min(D_i) \quad (7)$$

$$high = \max(D_i) \quad (8)$$

$$medium = (low + (high - low) / 2) \quad (9)$$

$$\sigma = high/8$$

$$m1 = high \quad (10)$$

$$m2 = m1 + (m1 * (FOU), \text{ where } FOU \text{ is in } (0,1)) \quad (11)$$

The fuzzy rules are an important part of a fuzzy system, for this proposed method the fuzzy rules consider various combinations of four gradients inputs D_i to produce only one gradient magnitude output. Fuzzy rules are shown in Table I.

TABLE I THREE FUZZY RULES FOR EDGE DETECTION [33].

Fuzzy Rules
1. If (D1 is HIGH), or (D2 is HIGH), or (D3 is HIGH), or (D4 is HIGH), then (E is Edge.)
2. If (D1 is MEDIUM), or (D2 is MEDIUM), or (D3 is MEDIUM), or (D4 is MEDIUM), then (E is Edge.)
3. If (D1 is LOW), (D2 is LOW), (D3 is LOW) and (D4 is LOW), then (E is Background.)

The first rule checks the four gradients inputs (D1, D2, D3, and D4) if it is high this means an edge. The second rule checks the four gradients inputs (D1, D2, D3, and D4) if it is medium this means also an edge. The third rule is only to confirm the first two because if the four directions are low this means there is no edge in this pixel [6] [33].

VII. PROPOSED METHOD

In the preceding research [6], we have proposed a novel LSB Steganography and fuzzy logic, which used gradient type-1 fuzzy logic edge detection technique to increase edge pixels, to embed more secret data into the edge pixels than the non-edge pixels, based on the (LSB) substitution technique. In this proposed scheme we extend the scheme in [6] by applying type-2 fuzzy logic system T2FLS to increase edge pixels which means increase payload and very good (PNSR values) quality of stego image.

VIII. EXPERIMENTS AND RESULTS

In this section, experimental results are performed to demonstrate the performance of our proposed method. We use the maximum number of embedded bits per pixel (bpp) to measure the embedding capacity (payload). Its formula is defined as follows:

$$bpp = \frac{\text{Maximal Embedding bits}}{H \times W}, \tag{14}$$

where H and W, respectively, are the height and width of the original cover image.

We use two viewpoints to measure the quality of stego image. The first one is the peak signal-to-noise ratio (PSNR) metrics, which calculate the difference between the stego and cover images, where a higher PSNR means better quality than the stego image.

In the second one, we evaluate the quality of the stego image against that of the cover image, as seen by the human visual system (HVS). PSNR formula is defined as follows:

$$PSNR = 10 \cdot \log_{10} \left(\frac{255^2}{MSE} \right) (dB), \tag{12}$$

where the (MSE) is the mean square error between the cover image and stego image. For a cover image with width W and height H, the MSE formula is defined as follows:

$$MSE = \sum_{i=1}^H \sum_{j=1}^W (p_{ij} - \hat{p}_{ij})^2 / (H \times W) \tag{13}$$

To conduct our experiments, we used, respectively, six 128×192 RGB color images from the BSD300 dataset, three from the training images, and three from the testing, as shown in Fig. 10. We also used the ‘Lena’ image with size 128×128 to compare our results with previous studies. ‘Lena’ image is shown in Figure 9.





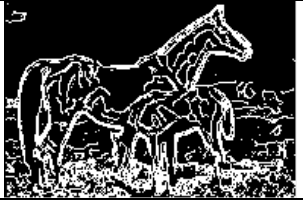
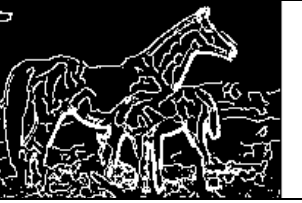











Fig. 9 ‘Lena’ image

		
3096	42049	253027



Fig. 10 Six128×192 images from BSD300 dataset

Canny	Hala et al's T1FS	Our proposed T2FS
		
2008	2700	3037
		
3117	5717	4326
		
3874	6467	7250
		
1995	2600	2735
		
1624	3432	4568

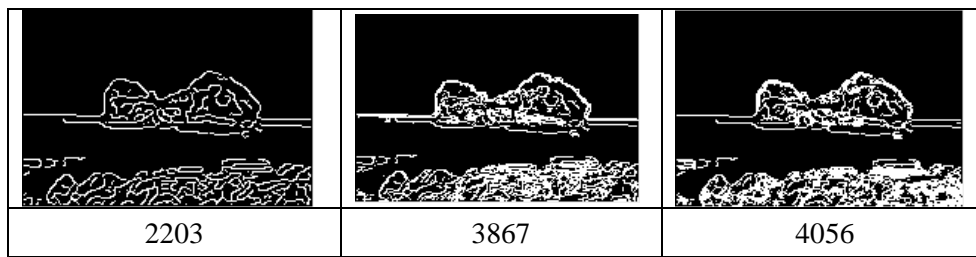











Fig. 11 The number of edge pixels detected by Canny, Hala et al.'s T1FLS [[HYPERLINK \l "Hal20" 6](#)], and our T2FLS edge detector.

Canny edge detection		
		
Image name = 3096	Image name = 3096	Image name = 3096
PSNR = 48.4630 dB	PSNR = 45.0340 dB	PSNR = 44.5054 dB
payload = 1.3301 bpp	payload = 2.6680 bpp	payload = 3.0010 bpp
Ratio = 32688 bits	Ratio = 65568 bits	Ratio = 73752 bits
		
Image name = 113044	Image name = 113044	Image name = 113044
PSNR = 48.1963	PSNR = 44.2973	PSNR = 42.7058
payload = 1.3339	payload = 2.6679	payload = 3.6337
Ratio = 32784	Ratio = 65568	Ratio = 89304
Hala et al.'s T1FLS edge detector [6]		
		
Image name = 253027	Image name = 253027	Image name = 253027
PSNR = 46.4572	PSNR = 42.2049	PSNR = 40.1558
payload = 1.3301	payload = 2.66	payload = 4.5488
Ratio = 32688	Ratio = 65568	Ratio = 111792









		
Image name = 253036	Image name = 253036	Image name = 253036
PSNR = 48.1633	PSNR = 44.5028	PSNR = 43.3270
payload = 1.3339	payload = 2.6679	payload = 3.3339
Ratio = 32784	Ratio = 65568	Ratio = 81936
Proposed T2FLS edge detector		
		
Image name = 42049	Image name = 42049	Image name = 42049
PSNR = 48.3825	PSNR = 42.9341	PSNR = 41.4912
payload = 1.3301	payload = 73608	payload = 98424
Ratio = 32688	Ratio = 2.9951	Ratio = 4.0049
		
Image name = 249061	Image name = 249061	Image name = 249061
PSNR = 49.4067	PSNR = 44.8596	PSNR = 44.0657
payload = 1.3301	payload = 2.6680	payload = 3.0010 bpp
Ratio = 32688	Ratio = 65568	Ratio = 73752

Fig. 12 Experimental results of the proposed T2FLS edge detector Comparison with canny and hala et al.'s T1FLS [6] edge detection using BSD300 dataset image.

TABLE II EXPERIMENTAL RESULTS OF THE PROPOSED SCHEME WITH DIFFERENT VALUES OF FOU ON 'LENA' IMAGE WITH THE SIZE OF 128x128

FOU = 0.2		FOU = 0.4		FOU = 0.6		FOU = 0.8	
PSNR	Payload	PSNR	Payload	PSNR	Payload	PSNR	Payload
48.0248	1.1255	47.8706	1.1255	47.8455	1.1255	47.8557	1.1255
42.4300	3.0965	42.3782	3.0965	42.3229	3.0965	42.3904	3.0965
40.7699	4.4973	40.7028	4.4891	40.6151	4.4973	40.5643	4.4973
-	4.5681	40.5463	4.5681	40.5610	4.5643	40.5026	4.5681

TABLE III EXPERIMENTAL RESULTS OF THE PROPOSED SCHEME COMPARISON WITH PREVIOUS STUDIES ON 'LENA' IMAGE WITH THE SIZE OF 128×128

J. Bai et al's Fuzzy edge detector [7]		Hala et al.' s T1FLS edge detector [6]		Proposed T2FLS edge detector	
PSNR	Payload	PSNR	Payload	PSNR	Payload
47.003	1.793	47.9370	1.223	48.0248	1.1255
34.554	3.586	42.2255	3.515	42.4300	3.0965
26.308	4.586	41.5490	4.0776	41.1389	4.0776
-	-	-	-	40.7664	4.5000

In this section, the experiments results of the proposed T2FS method will be presented and discussed by applying different edge detectors, which are shown in Figure 12. In each experiment, the performance of the stego image is measured by using the PSNR and HVS.

The experiments were applied on images from BSD300 dataset which are shown in figure 10. Figure 11 illustrates the proposed T2FS edge detector compared with T1FS and canny edge detectors. From this comparison, we can see that the proposed T2FS edge detector has a larger number of edge pixels.

From Figure 12, the best PSNR value obtained by the canny edge detector is 48.4630 when the capacity reached to the value 1.3301 bpp, and the worst PSNR obtained is 40.7058, when the capacity reached to the value 3.6337 bpp. For the Hala's scheme the best PSNR obtained is 48.1633 when the capacity reached to the value 1.3339 bpp, and the worst PSNR obtained is 40.1558, when the capacity reached to the value 4.5488 bpp. The best PSNR obtained by the proposed T2FS scheme is 49.4067 when the capacity reached to the value 1.3301 bpp, and the worst PSNR obtained is 41.4912, when the capacity reached to the value 4.0049 bpp.

Table 2 lists the different payloads and PSNRs values according to the change of FOU's values achieved by the proposed T2FS edge detector on 'Lena' image. This difference in values helps us to understand how the changes in the FOU affect the results of proposed edge detector. These changes can expound as follows: the different values of FOU represent various levels of uncertainty, and through this, we can compare the best PSNRs values and then determine the optimal FOU level for modeling the image. In this example the best PSNR value was the one obtained with the FOU = 0.2.

Table 3 presents the performance in terms of payload and PSNR with two previous schemes [6] [7] on 'Lena' image, with the size of 128×128. The results of comparison demonstrated that the PSNR values achieved by our proposed scheme improved that the quality of stego image in comparison with the previous schemes in [6] [7].

I. CONCLUSIONS AND FUTURE WORK

In this paper, we have proposed an edge detection method based on the gradient technique combines type-2 fuzzy with the canny edge detector to increase the payload while

maintaining quality of stego image. We used the (LSB) substitution technique to embed the secret data into the cover image. Various experiments have been done which compared with the other previous schemes, according to the PSNR results of all experiments we can conclude that our proposed scheme provides better stego image quality and higher embedding capacity than previous schemes.

REFERENCE

- [1] Kumar, P., & Sharma, V. K. Information Security Based on Steganography & Cryptography Techniques: A Review. *International Journal*, vol.4, no.10, pp.246-250, 2014.
- [2] Kaur, A., Dhir, R., & Sikka, G. A New Image Steganography Based On First Component Alteration Technique. *International Journal of Computer Science and Information Security(IJCSIS)*, vol.6, no.3, pp.53-56, 2010.
- [3] Dadgostar, H., & Afsari, F. Image steganography based on interval-valued intuitionistic fuzzy edge detection and modified LSB. *Journal of information security and applications* vol.30, pp.94-104, 2016.
- [4] Morkel, T., Eloff, J. H., & Olivier, M. S. An Overview of Image Steganography. *Information and Computer Security Architecture (ICSA) Research Group*, vol.1, no.2, 2005.
- [5] Ioannidou, A., Halkidis, S. T., & Stephanides, G. A novel technique for image steganography based on high payload method and edge detection. *Expert systems with applications*, vol.39, no.14, pp.11517-11524, 2012.
- [6] Yusuf, H. S., & Hagra. High Payload Image Steganography Method Using Fuzzy Logic and Edge Detection. *International Journal of Computer Science Trends and Technology (IJCT)*, vol.8, no.4, pp.123-133, 2020.
- [7] Bai, J., Chang, C. C., Nguyen, T. S., Zhu, C., & Liu, Y. A high payload steganographic algorithm based on edge detection. *Displays*, vol.46, no.10, pp.42-51, 2017.
- [8] Gonzalez, C. I., Melin, P. & Castillo, O. Edge detection method based on general type-2 fuzzy logic applied to color images. *Information*, vol.8, no.3, p.104, 2017.
- [9] Zadeh, L. A. Information and Control. *Fuzzy sets*, vol.8, pp.338-353, 1965.
- [10] Yusuf, H. S., & Hagra, H. Towards Image Steganography Using Type-2 Fuzzy Logic and Edge Detection. In *2018 10th Computer Science and Electronic Engineering (CEECE)*.IEEE, pp. 75-78, 2018.
- [11] Hellmann M. Fuzzy Logic Introduction. *Université de*

Rennes, vol.1,2001

- [12] Mohammed, H. A., & Hagrass, H. Towards Developing Type 2 Fuzzy Logic Diet Recommendation System for Diabetes. In *2018 10th Computer Science and Electronic Engineering (CEECE).IEEE*, pp.56-59, 2018.
- [13] Hagrass, H., Callaghan, V., Colley, M., & Carr-West, M. A fuzzy-genetic based embedded-agent approach to learning and control in agricultural autonomous vehicles. In *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No. 99CH36288C)*, vol.2, pp.1005-1010. IEEE, 1999.
- [14] Starkey, A., Hagrass, H., Shakya, S., & Owusu, G. A multi-objective genetic type-2 fuzzy logic based system for mobile field workforce area optimization. *Information Sciences*, pp.390–411, 2016.
- [15] Alhassan, M. S. E., & Hagrass, H. A Congestion Control Approach Based on Weighted Random Early Detection and Type-2 Fuzzy Logic System. *International Journal of Computer Science Trends and Technology (IJCT)*, vol.8, no.4, pp.83-94, 2020.
- [16] Bernardo, D., Hagrass, H., & Tsang, E. A genetic type-2 fuzzy logic based system for the generation of summarised linguistic predictive models for financial applications. *Soft Computing*, vol. 17, no. 12, pp. 2185-2201, 2013.
- [17] Sakalli, A., Kumbasar, T., Yesil, E., & Hagrass, H. Analysis of the performances of type-1, self-tuning type-1 and interval type-2 fuzzy PID controllers on the magnetic levitation system. In *2014 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pp. 1859-1866, IEEE, 2014.
- [18] Callaghan, V., Colley, M., Hagrass, H., Chin, J., Doctor, F., & Clarke, G. Programming iSpaces—A tale of two paradigms. In *Intelligent Spaces*, pp. 389-42, 2006.
- [19] Hagrass, H. A hierarchical type-2 fuzzy logic control architecture for autonomous mobile robots. *IEEE Transactions on Fuzzy systems*, vol. 12, no. 4, pp. 524-539, 2004.
- [20] Lynch, C., Hagrass, H., & Callaghan, V. Embedded interval type-2 neuro-fuzzy speed controller for marine diesel engines. *Proceedings of the International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU 2006)*, pp. 1340-1347, 2006.
- [21] Antonelli, M., Bernardo, D., Hagrass, H., & Marcelloni, F. Multiobjective evolutionary optimization of type-2 fuzzy rule-based systems for financial data classification. *IEEE Transactions on Fuzzy Systems*, vol.25, no. 2, pp. 249-264, 2016.
- [22] Zadeh, L. A. Fuzzy Sets As A Basis For A Theory Of Possibility. *Fuzzy Sets and Systems*, vol.1, no.1, pp. 3-28, 1978.
- [23] Karnik, N. N., & Mendel, J. M. Operations on type-2 fuzzy sets. *Fuzzy sets and systems*, vol.122, no.2, pp.327-348, 2001.
- [24] Liang, Q., Karnik, N. N., & Mendel, J. M. Connection admission control in ATM networks using survey-based type-2 fuzzy logic systems. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol.30, no.3 pp. 329-339, 2000.
- [25] Mendel, J. M., & John, R. B. Type-2 fuzzy sets made simple. *IEEE Transactions on fuzzy systems*, vol.10, no.2, pp.117-127, 2002.
- [26] Andreu-Perez, J., Cao, F., Hagrass, H., & Yang, G. Z. A self-adaptive online brain-machine interface of a humanoid robot through a general type-2 fuzzy inference system. *IEEE Transactions on Fuzzy Systems*, vol.16, no.1, pp.101-116, 2016.
- [27] Hagrass, H., & Wagner, C. Towards the wide spread use of type-2 fuzzy logic systems in real world applications. *IEEE Computational Intelligence Magazine*, vol.7. no.3, pp.14-24, 2012.
- [28] Alhassan, M. S. E., & Hagrass, H. Towards Congestion Control Approach Based on Weighted Random Early Detection and Type-2 Fuzzy Logic System. In *2018 10th Computer Science and Electronic Engineering (CEECE)*, pp. 71-74. IEEE, 2018.
- [29] Bernardo, D., Hagrass, H., & Tsang, E. A genetic type-2 fuzzy logic based system for financial applications modelling and prediction. In *2013 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*: pp. 1-8. IEEE, 2013.
- [30] Yao, B., Hagrass, H., Alhaddad, M. J., & Alghazzawi, D. A fuzzy logic-based system for the automation of human behavior recognition using machine vision in intelligent environments. *Soft Computing* vol.19, no.2, pp.499-506, 2015.
- [31] D. Bernardo, H. Hagrass, E. Tsang, “A genetic type-2 fuzzy logic based system for the generation of summarised linguistic predictive models for financial applications”, *International Journal of Soft Computing*, vol.17, no.12, pp. 2185-2201, August 2013.
- [32] Bilgin, A., Hagrass, H., Malibari, A., Alhaddad, M. J., & Alghazzawi, D. Towards a linear general type-2 fuzzy logic based approach for computing with words. *Soft Computing*, vol.17, no.12, pp.2203-2222, 2013.
- [33] Jain, N., Meshram, S., & Dubey, S. Image steganography using LSB and edge-detection technique. *International Journal of Soft Computing and Engineering (IJSCE)*, vol.2, no.3, pp.217-222, 2012.
- [34] Melin, P., Mendoza, O., & Castillo, O. An improved method for edge detection based on interval type-2 fuzzy logic. *Expert Systems with Applications*, vol.37, no.12, pp.8527-8535, 2010.
- [35] Chen, W. J., Chang, C. C., & Le, T. H. N. High payload steganography mechanism using hybrid edge detector. *Expert Systems with applications*, vol.37, no.4, pp.3292-3301, 2010.
- [36] Kaur, E. K., Mutenja, V., & Gill, I. S. Fuzzy logic based image edge detection algorithm in MATLAB. *International Journal of Computer Applications*, vol.1, no.22, pp.55-58, 2010.