Iranian Journal of Fuzzy Systems

Volume 18, Number 1, (2021), pp. 101-116

Original paper

Hybrid data compression using fuzzy logic and Huffman coding in secure IOT

S. Nosratian[1], M. Moradkhani[2] and M. B. Tavakoli[3]

[1,3] *Department of Electrical Engineering, Arak Branch, Islamic Azad University, Arak, Iran*
[2] *Department of Electrical Engineering, Ilam Branch, Islamic Azad University, Ilam, Iran*

nosratiansajad@gmail.com, moradkhani.m@ilam-iau.ac.ir, mb-tavakoli@iau-arak.ac.ir

Abstract

Research in the area of Internet of things (IoT) and cloud computing has gained a considerable attention in today's world of information technology (IT). Data compression and security are increasingly appreciated due to their imperative roles in online data sharing and transfer in multimedia networks. Hence, image compression is focused on reducing data redundancy and saving memory and transmission bandwidth. Because of the increased entropy of coded images, it is difficult to reduce the amount of lossless information after encryption using the existing methods. Accordingly, using the hybrid Huffman coding algorithm, it is possible to guarantee the security and verification of information. Therefore, an efficient compression method is proposed that combines fuzzy logic with Huffman coding. In this method, coding is based on Hoffman code, which is statistically independent and defines fuzzy logic-based weighting functions for the frequency of the existing symbols in data to generate efficient code for compression. Fuzzy logic is then coded to provide a secure information mapping on the information codes. In this proposed system, the key information is coded using the special key of that information and the data is embedded using the data encryption key and data in the receiver are decoded using the same key. Here, different compression techniques are compared with Huffman coding. The simulation is performed under matlab2017b software. The results show that a good compression rate is achieved, there is no significant difference between the decoded and original images and compression ratio in the image is increased to over 40%.

Keywords: Huffman coding, fuzzy logic, secure information mapping, IoT, information technology, compression ratio.

1 Introduction

Recently, Internet of Things (IoT) has been a subject of considerable contemporary research for its extensive applications in wireless sensor network and remote monitoring. IoT is a model that includes ordinary people and has the ability to communicate with other devices via the Internet [4]. As the broadband and high-speed internet is available to public and its connection cost is reduced, more devices and sensors could connect it [30]. Such conditions provide a good basis for IoT growth. There is a lot of complexity around IoT because people want to access and monitor every object from anywhere in the world [21]. The key issue in IoT is to secure the data collected from Internet-connected objects. Data security can be combined with authentication, encryption and decryption, message authentication codes, Hash functions and digital signatures and other items provided [18].

Adding encryption to compression algorithms is an interesting proposition that can lead to system security and hybrid compression while increasing system performance simultaneously. The main challenge is to design safe and lossless systems in compression performance. Entropy encryption algorithms such as Huffman coding and arithmetic encryption [17] are used not only for lossless compression but also as part of discrete compression algorithms such as the Joint Photographic Experts Group (JPEG) [29]; therefore, they lead to the security of larger systems. Data compression is a common need for most computer applications. Data compression is an important application in file storage and

Corresponding Author: M. Moradkhani
Received: September 2019; Revised: May 2020; Accepted: August 2020.

distributed systems. Data compression is used in the field of multimedia, text documents and database tables. Data compression methods can be categorized by several methods. One of the most important criteria for classification is whether compression algorithms delete part of the data that cannot be retrieved during compression.

Lossy data compression:
An algorithm that deletes part of the data is called lossy data compression. The lossy data compression algorithm is usually used when complete compatibility with the original data is not required after the compression is removed. For example, lossy data compression includes video or video data compression.

Lossless data compression:
Lossless data compression technique is used by methods to retrieve all the data that is compressed without any damage or loss. This method is used in cases where compressed data is required as it is not compressed without any deficiency [5]. Lossless data compression is utilized, for example, in law enforcement in text files, database tables and medical imaging. Lossless data compression algorithm is proposed and applied for various data. Some of the main techniques include Huffman coding, Run Length Enoding, arithmetic coding and dictionary-based coding.

In secure IoT, lossless compression is usually used due to the high level of security of information. The data compression algorithm commonly used for image compression results in lower quality of system security. Compression is achieved by removing duplicated bits or extra bits from the image or data. There are many data compression algorithms considered as universal compression algorithms that can compress almost any data type. According to the existing methods, not all the details of the compressed data can be retained. Therefore, using smart techniques along with compression algorithms can be useful. For example, in [23], using discrete cosine transform and fuzzy logic techniques, an image compression technique is proposed. The proposed method improves system performance in both compression ratio and image comprehension, but in this method, image quality is reduced after image extension and there is no complementary encryption technique to enhance system security. In [27], a lightweight encryption algorithm is proposed as Secure IoT (SIT). This is a 64-bit block encryption method and requires a 64-bit key to encrypt the data. The algorithm architecture is a combination of feistel and a uniform substitution-permutation network. The simulation results show that the algorithm offers significant security only in five rounds of encryption. The hardware implementation of the algorithm is performed on a low-cost 8-bit microcontroller, and the results of the code size, memory usage, and encryption / decryption execution cycles are compared with other existing encryption algorithms. The problem with this method is the complexity of execution, the duration of the encryption and decryption operations, which can work at an ideal speed on the network. In [28], a new encryption program is proposed with Huffman coding to encrypt data transferred by the multimedia network. The suggested method manipulates compressed data based on Huffman encryption without changing the ratio of compression. The encryption is performed through the mutation procedure that briefly produces multiple Huffman tree mutations by switching the Hoffman main tree branches i.e. 0 to 1 and 1 to 0 utilizing a checklist rotation method. The proposed method offers a good technique to improve the security of data transmission. Instead, the cost of design and complexity make the implementation of the proposed design to perform this idea.

If a wireless sensor network (WSN) is integrated into the Internet as part of IoT, new security challenges such as setting up a secure channel between the sensor node and an Internet host will emerge. In order to establish secure communication, a Closed Frequent Huffman (CF-Huffman) based hybrid signcryption technique is developed in [20]. The proposed Huffman technique helps to improve the compression ratio based on the frequency of data items. Hybrid signcryption is based on KEM and DEM techniques. The KEM algorithm uses the KDF technique to encapsulate the symmetric key. The DEM algorithm uses the AES algorithm to encrypt the original message. The signature is done by the secure hash function. Eventually the coded data is compressed with Huffman text coding and end this technique in a security and compression function.

The IoT technologies and cloud computing are integrated to form a platform called IoTcloud. As these technologies are widely used, there is a need to secure the data collected from connected devices through these technologies. [11] presents a new data security approach on IoTcloud. The proposed method provides data security using the ciphers generated by the Huffman algorithm and DNA encryption. It provides two-level data security and emphasizes the use of the Huffman coding algorithm to generate the key and use DNA encryption for data security. The proposed method uses the symmetric key encryption. In [24], the coding program uses two algorithms i.e. Blowfish algorithm and Hoffman compression algorithm to generate PDF, DOC, XLS and text types. The Huffman compression algorithm is used to minimize size after encryption with the Blowfish algorithm. As the proposed coded program enters the network without a password key, it cannot be opened or restored during encryption. The encryption process time is faster than the decryption process, and with this encryption program, the process of storing and exchanging information becomes more secure. However, the problem of speed in this method and key detection in the decoding section is a fundamental problem in this proposed method. In the proposed system presented in [19], the original image is encoded using the Huffman coding key and the data is embedded in the data using the encryption key. Then the image and data are decoded at the receiver side using the corresponding key. In the proposed method, the keys are part of the information level when hiding secret information. Accordingly, a large empty container is obtained and the data memory can insert more hidden messages into the coded image. This method guarantees a high level of security for hiding information, but still wastes high encryption time.

In [3], an efficient compression method is proposed that combines fuzzy logic with Huffman coding. When the image pixel is normalized, each value of the image pixel relating to that image preview is specified and interpreted. The image is divided into pixels, which are specified by a pair of approximation sets. In this method, coding is based on Hoffman code, which is statistically independent to produce efficient code for compression and decoding is based on rough fuzzy logic used for image pixel reconstruction. The technique utilized here is the rough fuzzy logic with Huffman coding algorithm (RFHA). In this method, a high compression rate is achieved and visually there is little difference between the compressed and original images. The problem with this method is the lack of a secure structure for the proposed method that creates free access to information for the public. Finally, the article [25] is a review study. In this paper, image compression methods such as DCT, DWT and Hybrid are discussed. Hybrid image compression is done by DWT and DCT, which performs discrete cousin transform on discrete wavelet conversion coefficients. Compression techniques are useful for compressing an image for high values of PSNR (peak signal to noise ratio) and CR (compression ratio) [7, 31]. These methods take advantage of any compression method known as hybrid compression. This new hybrid technique minimizes various compression problems. According to this work, the fuzzy-based hybrid image compression technique offers higher compression than normal hybrid image compression. The fuzzy logic method can also be used to calculate and modify coefficients and also improve the accuracy of compression. Therefore, in the present study, for the first time, a new technique is proposed to achieve information compression algorithm, which combines fuzzy logic and Huffman coding. Huffman coding is a well-known algorithm for generating minimum redundancy code compared to other algorithms. Improved Huffman coding with fuzzy logic-based intelligent encryption technique has been used in text, image and video compression [9, 16]. Approaches based on fuzzy relation equations and fuzzy transformations have been recently reported in the literature [10, 12]. However, in this method, a secure encryption technique based on fuzzy logic and compression coefficients of Huffman method is proposed in terms of fuzzy logic. It is important to note that, based on the available results the fuzzy and rough set synergy is for more secure data transfer, better compression with high ratio and very low availability.

As mentioned earlier, Huffman encoding involves a lossless encoding-based compression. When information is transmitted, this type of coding enables one-time reading and adaptation to changing conditions without the knowledge of the source distribution. The advantage of reading once is that the source can be encoded in an instant. Also, this encoding method allows multi-step compression for input data codes. Therefore, for a secure encryption method that is the purpose of the present work, compressed code mapping will occur during the different stages regardless of the image or text message signal content. This method can result in a completely random way by moving the resulting compressed codes through the coding path of an encryption without the possibility of identification for outsiders. The proposed design, of course, uses compression over two stages, which can be extended to further steps. Moreover, considering secure compression-based encryption, two compression steps have been used which increase the compression action over the single-stage mode. On the other hand, in the field of security, we have been able to achieve a compressed code transfer method based on a nonlinear model using fuzzy logic technique which greatly reduces the possibility of decoding without having specific information. Besides, the specific information is changed according to the content of the original data (both the image and the text). In the other words, as the content of the message changes, specific information is changed for decoding.

2 Fuzzy logic theory

Fuzzy set theory has been extensively used for modeling human thinking concepts and refers to the uncertainty in the information accessible to make decisions on the basis of multiple criteria. Replacement competence against criteria and the significant weight of criteria in relation to the expressed linguistic values is published by numbers. In the fuzzy set, to describe the fuzzy conditions, linguistic variables convert the linguistic variables into numerical variables and the real logical values are replaced by single intervals in the decision process [2]. Hence, a fuzzy set is considered as a finite, infinite, or infinite countable set of components, mathematically. In each case, each element is a member of a set or not. However, in fuzzy systems, the element may be part of or outside the set. Therefore, the answer to the question "X in a set A" has no definite right or wrong answer. Figure 1 shows the block diagram of the fuzzy logic diagram.

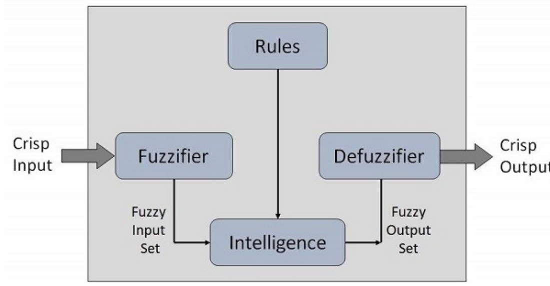


Figure 1: Block diagram of fuzzy logic diagram

2.1 Fuzzy set

A fuzzy set A is defined in global discourse, which is specified by a membership function $\mu_A : U \to [0, 1]$ given by Eq. (1).

$$A = \{(x, \mu_A(x))|x \in U \Lambda \mu_A(x) \in [0, 1]\}. \tag{1}$$

For each x member of the set A, $\mu_A(x)$ denotes the degree of relation x in A.

$$x \in (A, \mu) \Longleftrightarrow x \in A \Lambda \mu(x) \neq 0. \tag{2}$$

In addition, using the membership function, each element x as a member of U expresses a degree of relationship in each set A, indicating how much the element x may belong to set A. Therefore, the zero degree relationship member means that the element is not included in the fuzzy set; while a 1st degree element is fully included in the set.

2.2 Fuzzification

Assuming the feature of the application area relevant to the present work, for example, the triangular membership function is adopted. A triangular fuzzy number A can be set by three numbers (a, b, c) with a matching function by Eq. (3)

$$\mu(x) = \begin{cases} 0, & \text{Se} x < a, \\ \frac{x-a}{b-a}, & \text{Se} a \leq x \leq b, \\ \frac{c-x}{c-b}, & \text{Se} b \leq x \leq c, \\ 0, & \text{Se} c < x. \end{cases} \tag{3}$$

2.3 Fuzzy inference

Defuzzification is a process that creates a small amount and a size in fuzzy logic, meaning that fuzzy numbers are converted to a single number by different methods. This work expresses the maximum mean weight as a Eq. (4).

$$Z_0 = \frac{\sum \mu(x)i \times Wi}{\sum \mu(x)i}. \tag{4}$$

Given the Eq. (4), Z_0 is the defuzzification output, $\mu(x)i$ is the degree of association with the fuzzy set and w_i is the value of the fuzzy output weight.

3 Huffman coding

Huffman coding is a successful compression method that is initially used for text compression. Huffman's idea is that instead of using a fixed-length code such as 8-bit ASCII or DBCDIC for each symbol, a character in a source represents a shorter encoder and represents a shorter code that usually happens [26]. Huffman coding is considered as a lossless data compression technique to compress information by an entropy encryption algorithm [15]. In Huffman code, a variable-length code is utilized to encode a source symbol (like a character in a file) in which it has been introduced with respect to the estimated probability of the source symbol. Huffman code is based on the number of occurrences of a character. The basic of this particular approach is to use fewer bits to encrypt the data that are more likely to occur [22]. The average length of a Huffman code is influenced by the statistical iteration that generates the source of each symbol and generates the symbol from its alphabet or equivalent numeric value. In the Huffman Code Dictionary, each data symbol is associated with a coding word, so that no word code in the dictionary is similar to any other word [14]. The origin of Huffman encoding algorithm is a code tree. It is worthy to note that regarding to mentioned code tree, short code words are attributed to the frequent items and long code words to symbols which are scarcely utilized for both DC and AC coefficients (each one is assigned to the main component of the transferred data with variable length codes from the Huffman table set). Huffman codes should be introduced as inputs to network node coders. It is worth mentioning that the form contained in the Huffman tables in the data path is a profile of indirect decoder which must build its previous tables for data expansion [29]. Figure 2 depicts the flowchart of the Huffman algorithm. To illustrate the concept of this algorithm, an example is provided. Suppose a list contains the 0, 2, 14, 136, and 222 symbols. Their frequencies are shown in Table 1. As shown in this table, the symbol 0 is repeated 100 times in the list. Figure 3 and Table 1 indicate the Huffman tree and their final code [13].

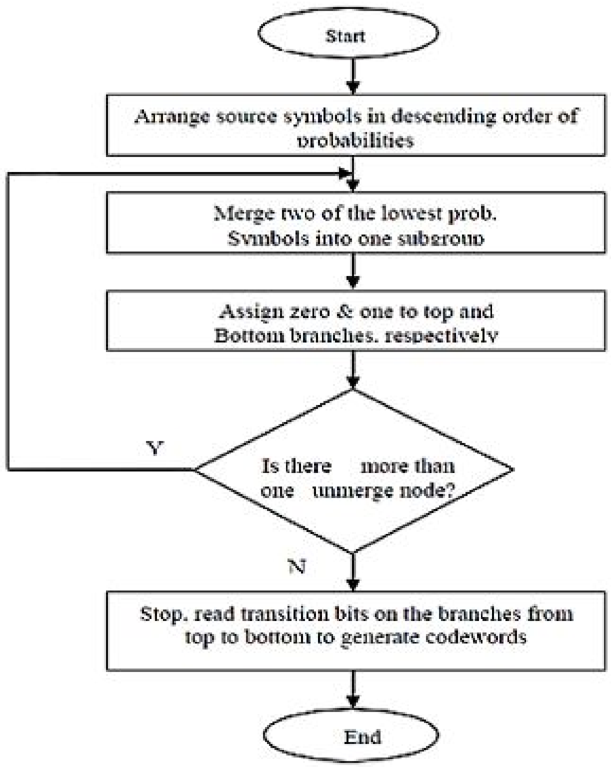


Figure 2: Huffman coding flowchart

Table 1: Displaying input symbols along with their frequency and codes sent to the decoder

Symbols	Code	Frequency
0	1	**100**
2	011	**011**
14	010	**9**
136	001	**7**
222	000	**5**

As depicted in Table 1, the lowest number of bits attributed to the largest frequency of symbols is one bit; therefore, bit 1 is allocated to the zero symbols. This means that bits of less than one bit cannot be assigned to this symbol. In recent years, data encryption has become more popular in network access. Providing security for the transmitted data plays an important role. The rapid development of wireless and wireless digital communications has made it possible to make extensive use of text data. However, there is very little research focusing on data encryption, conditional on data security and memory usage. Secure transfer of confidential information in public network space such as internet is one of the major challenges of today's communication. Watermarking is a branch of signal processing aimed at

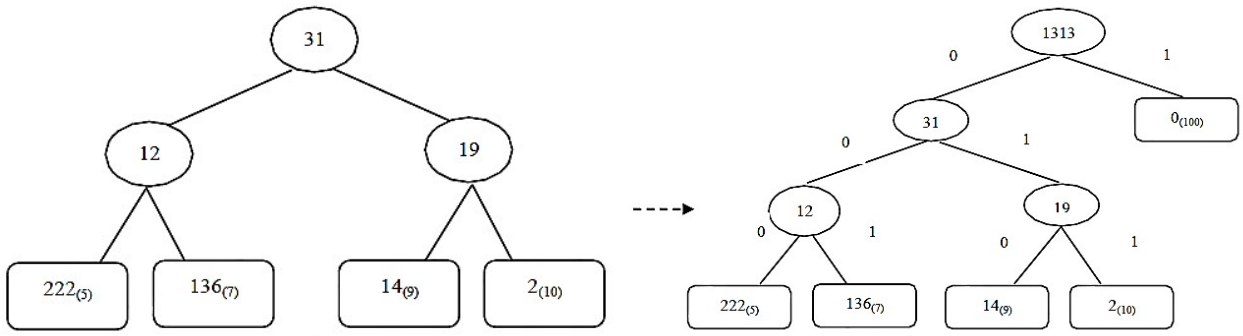


Figure 3: The Huffman tree construction process [13]

maintaining the confidentiality and availability of the message by location and frequency domain methods. Therefore, in this paper, the problem of data size optimization and data security problem for low bit rate transmission and high security is investigated based on Huffman coding. One of the advantages of using Huffman coding is that outsiders do not have access to data decoding, while no code can be a prefix for another code. Therefore, there will be no confusion in decoding process. On the other hand, in Huffman coding, weights are defined for the transferred data based on the frequency of data in the system, which varies depending on the data type of the Huffman code. Therefore, in the security domain, these weights functions are defined by a specific algorithm. In the current work, using a fuzzy logic algorithm, it is attempted to optimize the weighting functions for this coding. In this field, there are data that are transmitted in binary form and through classifying the data in m-bits, the data numbers are compressed under fuzzy optimized Hoffman code. In this research, using an optimized data encryption, the data is coded during compression and the system is protected from decoding these data by the hackers. The idea is used to encode data in the data transmission space on the network.

3.1 Encryption of proprietary information in the field of IoT

Cryptography comes from the Greek root of the words Kryptos and Graphin. Cryptography dates back to centuries ago. In the past, they used cryptography to protect messages that were exchanged between commanders, spies, diplomats, and so on. Cryptology is a science that discusses the issues of encryption and decryption. Cryptography is a process that converts a readable message into an unreadable format. Decryption encompasses activities related to code analysis and seeks to obtain the key and thereby obtain the original text from the code. Cryptography knowledge is one of the few branches of contemporary knowledge that derives its identity from human inability to solve mathematical problems. Encryption and decryption methods have always been an important factor in war strategies. Widespread American efforts to break the Japanese code during World War II are one of the most important tasks of the Alliance [8]. Security is a fundamental issue in modern communications. Many cybercrimes arise as the technology progresses. There are several solutions to prevent security risks. Such as:

- Turn off the unnecessary services
- Updating systems
- Reducing access to applications and reducing the number of users
- Using security protocols

However, another solution to address the security risks is the use of cryptography. Cryptography has two main components: algorithm and key. The algorithm is a mathematical converter or relation. The set of rules used in a coding process is called the encryption algorithm. A key is used to convert the original information. This information is unreadable and is known as coded message text. Since security is the main concern in the exchange of information, the encryption algorithms are an important part of it [1]. Most encryption algorithms used today are only concerned with security. Therefore, the performance is also essential for developing technologies. Today's common algorithms require high processing time and lack sufficient security. However, users who use a low bandwidth need an algorithm that has low processing power. Algorithms that have a high level of security have more processing power than low-level algorithms. Therefore, an optimal algorithm should be provided to provide users with security due to lower processing and higher security levels. Therefore, this article focuses on security and seeks to raise the level of security in data exchange using a new encryption algorithm. This is done by applying fuzzy logic in a Huffman encryption algorithm so that in this proposed algorithm, users will choose the key they want according to their needs. There are generally two types of algorithms: symmetric key algorithms and asymmetric key algorithms. In symmetric key algorithms, a key is used to encrypt and decode the message. Symmetric key encryption algorithms are referred to as secret key algorithms or private key algorithms. However, in asymmetric key algorithms, one key is used to encrypt the message and the other one is to decode it. Public key cryptography is an important class of asymmetric key algorithms. In these algorithms, the encryption key is usually called the public key, as it can be accessed without compromising the confidentiality of the message or the decryption key. The message decryption key is also called the private or secret key.

3.2 Fuzzy-Huffman hybrid encryption method

In this research, to achieve a secure coding, different techniques are used to enjoy real space with minimal processing speed and thus speeding up the coding for a low energy storage network such as wireless sensor network along with the highest security. According to the different techniques, making the key in this method is very simple and predetermined but with the help of mapping equations provided to a random code, a code is considered for each message by each receiver. Therefore, the proposed method is based on a symmetric key that can be changed per message and it depends on the inherent characteristics of the message and the data. In this part, the proposed approach is clarified in detail. An important feature of this method is that the message compression is performed optimally and intelligently and power and energy losses are reduced for transmission. However, by combining Huffman coding with fuzzy logic, a limit is achieved to encrypt the transmitted data, where Figure 4 represents a block diagram model for the proposed scheme that is introduced in accordance with the form for the encryption mode.

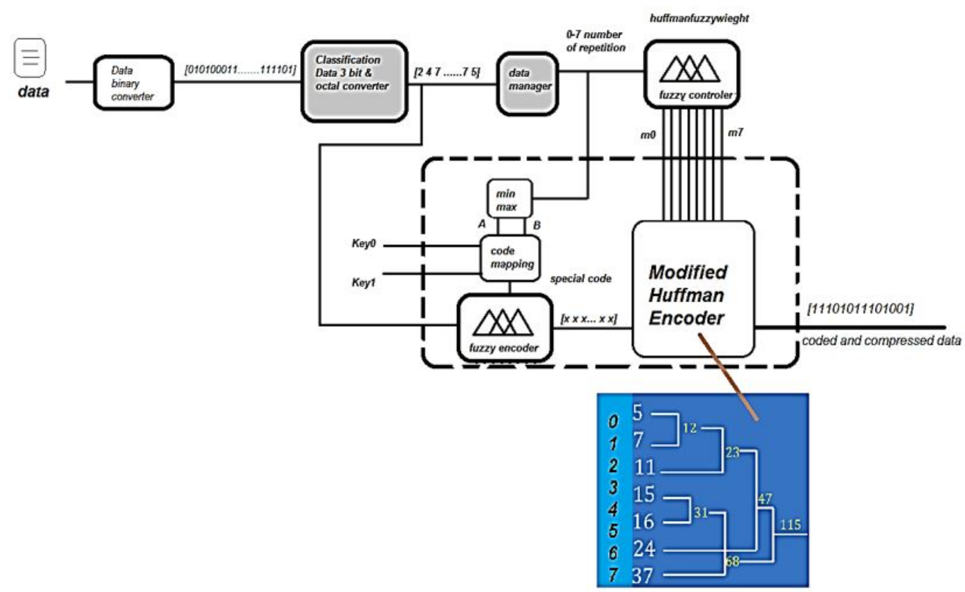
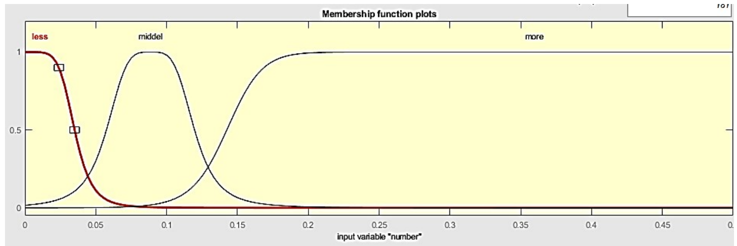


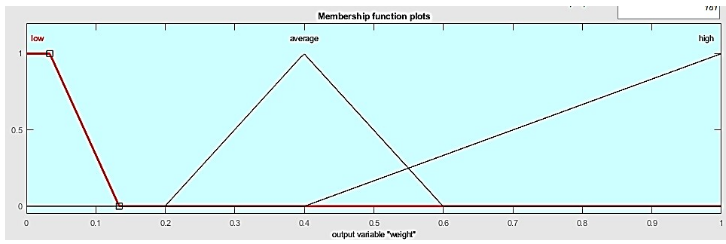
Figure 4: Block diagram of Huffman and fuzzy hybrid encoder

As shown in the Figure 4, the message or image information is initially loaded as an input. In this work, both text and image inputs are simulated. This information is converted into binary packets that are prepared for transmitting and processing of information. Therefore, from this point on, the binary data is dealt with. After converting the data to binary code, the binary data is converted to three-bit packets and then transferred into the octal from the binary space. Accordingly, there are data based on 0 to 7, for example the values of [2 4 7 $\cdots$ 7 5] are observed. After converting binary data to an octal base, the next block is achieved that shows the frequency of each number [0 1 2 3 4 5 6 7]. Using a fuzzy control, after normalizing the frequency, the optimal compression weight of the Huffman algorithm is computed and obtained. In this study, the fuzzy control is determined with Huffman fuzzy weight of Mamdani type in Figure 5. In this block, the input and output weight functions are presented in the form shown in fuzzy rules in Table 2.

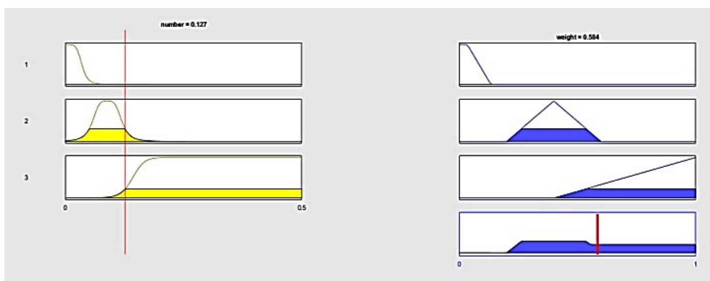
The obtained weight coefficients in this fuzzy function are used to compress the information and are sent to the network with the packet to be used by the receiver to retrieve the information.



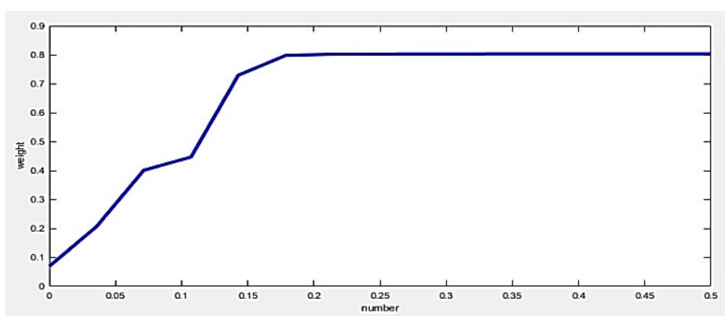
(a)



(b)



(c)

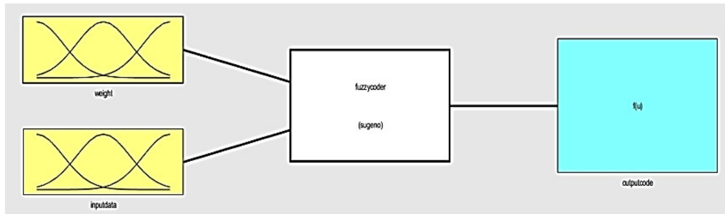


(d)

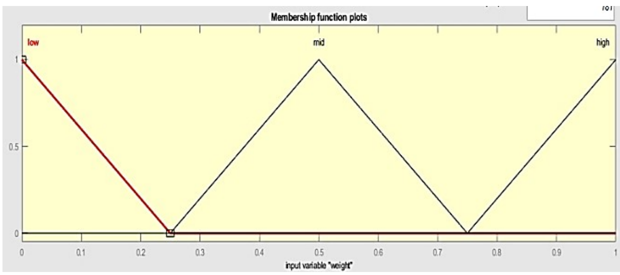
Figure 5: Huffman fuzzy weight fuzzy logic block for determining the weights of the Huffman algorithm, (a). Input Membership Functions, (b). Output membership functions, (c). Fuzzy rules, (d). The input-output characteristic curves of the system.

Table 2: Weight determination rules

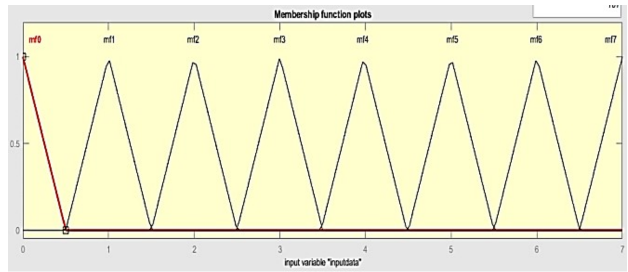
Repeat number	Wight
Less	Low
Middle	Average
More	High



(a)

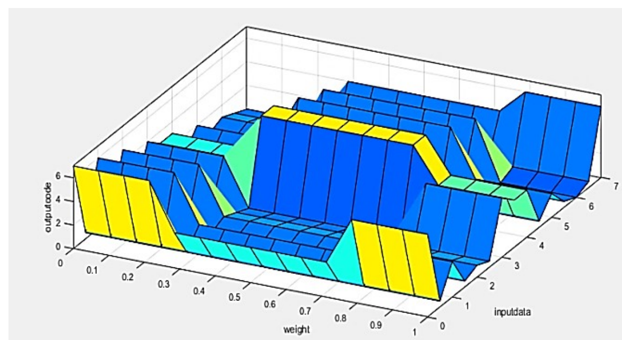


(b)



(c)

(d)



(e)

Figure 6: Fuzzy encoder fuzzy logic block for assigning different mappings according to the proprietary key, (a). Takagi-Sugeno system, (b). Input membership functions, (c). Output membership functions, (d). Fuzzy rules, (e). Output Input feature.

3.3 Cryptography under fuzzy mapping

After determining the weight for elements 0 to 7 to compress, it is time to propose a solution to encrypt data based on a complex encryption. Therefore, in this section, a mapping will be introduced that changes the octal code to its actual mapped value. For this part, two numeric keys are defined for the system as public keys. These keys are defined by the minimum values and the maximum number of frequencies obtained in the previous step, which are different values for each data as follows: (Eq. (5))

$$\text{Special code } = A * Key0 + B * Key1. \tag{5}$$

The values of A and B are the minimum and maximum values of 0 to 7, respectively, for the transmitted message or image and they vary for different data. The resulting key range is the numbers from zero to one, which will change with the values of A and B. In this case, a fuzzy encoder system is defined for this research, which introduces non-identifiable mappings for different values in the interval [0 and 1]. Table 3 presents the rules governing this structure. Figure 6 also shows the fuzzy encoder structure for the Takagi-Sugeno type and input membership functions. The important point in this proposed encryption method is to change the proprietary key for different message inputs so that the transmitter itself does not know the proprietary key. In this paper, three domains are selected for the value of the proprietary key. Accordingly, the number of mapping is increased by domains and the level of security is enhanced. The notable point in this research is that in selecting the mappings changing the numbers and number mapping is arbitrary, which is provided by a non-linear and unpredictable model that by increasing the proprietary key domain, the number of these non-linear mappings is increased.

Table 3: Fuzzy rules of mapping

InputW	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
Low	7	6	5	4	3	2	1	**0**
Mid	3	2	1	0	7	6	5	**4**
High	7	3	1	5	4	2	0	**6**

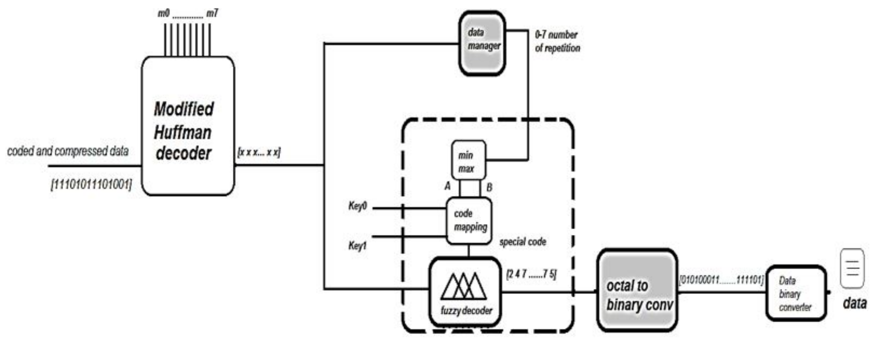


Figure 7: Block diagram of Hoffman and Fuzzy hybrid decoder

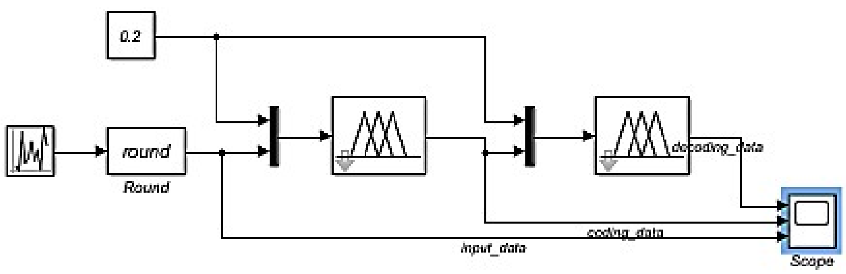
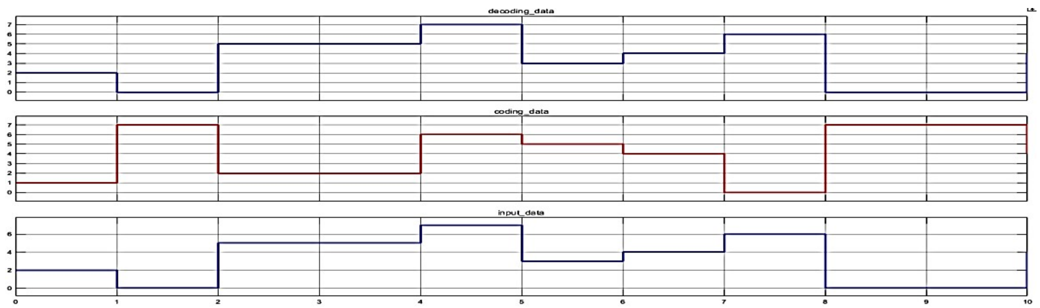


Figure 8: Simulink Display of the encoder and decoder system

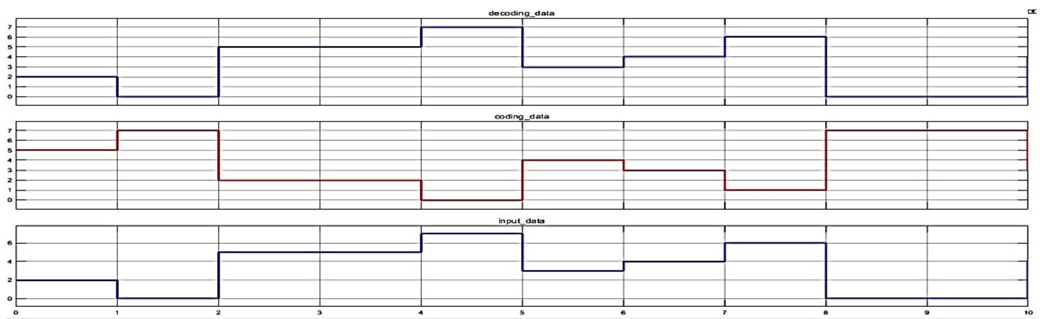
In this section, with the development of mappings, the cryptography complexity is used despite the simplicity and low processing of the system. After this step, the coded data is sent to the destination according to the defined path and is then the received code is decoded by means of a decoder in accordance with Figure 7. According to this figure, the reverse steps in the coding section are performed to obtain the packet. Therefore, in this structure, the received data together with the weights and the key of the code are used based on the steps shown in the figure to reach the original data code. In this set, the same key is used to obtain the key of the equation (5) with respect to the minimum and maximum values of 0 to 7. In these blocks, however, the decoders are used in the fuzzy and Huffman sections. For the encryption and decoding performance correction in accordance with Figure 8 and Figure 9, it is investigated and simulated for examples of data for different inputs and different proprietary key weights. After decoding, the resulting code is in the octal basis. In the last section, the conversion block based on the binary is used to calculate the different values. The output code is the binary code of the original data, which is eventually converted to the original data values including the image or text.

4 Displaying encryption results

In this section, the results are presented in two sections. In this case, two types of text and image data are used. Finally, the results for each section are examined and presented according to the mentioned coding program. In each example, various images and texts have been tested and simulated, which results in the correct performance and speed of information processing.



(a)



(b)

Figure 9: Output display of the encoder and decoder function, (a). Display of encryption and decryption for w=1, (b). Display of encryption and decryption for w=0.2.

Example 4.1. *(Textual data example)*
In the first example, the sent text is 'Hello, I'm a good student. sent by the proposed technique with the public key [10 and 3]. The information packet after one stage of encryption, converting into binary, 3-bit classification and octal code formulation are as follows:

0	0	0	4	1	2	6	0	7	4	2	2	0
	5	2	1	4	2	0	3	7	0	4	3	6
	6	6	4	4	6	4	0	2	2	4	7	2
	2	7	6	1	4	3	4	6	7	5	7	1
	4	1	6	6	3	1	1	7				

The information code after encryption under fuzzy mapping and the proposed keys are as follows.

7	7	7	4	3	1	0	7	6	4	1	1	7
	2	1	3	4	1	7	5	6	7	4	5	0
	0	0	4	4	0	4	7	1	1	4	6	1
	1	6	0	3	4	5	4	0	6	2	6	3
	4	3	0	0	5	3	3	6				

The coded code was finally obtained after compressing and re-coding the hybrid Hoffman-Fuzzy

The original code number to send before compression is 216 bits, which is reduced to 166 bits after compression, and this is for a single-sentence text that covers 27 characters that by assigning one byte (eight bits) to each character, the number of bits sent is greatly reduced. However, for longer texts, the difference between the original data size and the coded data is greatly increased. On the other hand, the mapping applied under the introduced key causes a change in the encryption structure of this text file, which will be uncoded without recognizing the proprietary key. Moreover, determining this proprietary key for different terms and texts with different characters even with applying the same public key so that the coder itself will not find the proprietary key for the text and the key will change by altering the

$$
\begin{array}{ccccccccccccc}
[0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 \\
 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\
 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \\
 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \\
 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\
 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\
 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\
 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\
 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 \\
 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\
 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0] & & &
\end{array}
$$

data. Now in the receiver, the data is decoded with the public key and the text features including the coded size, coded data and compression weight; after that, the data is obtained as follows.

$$
\begin{array}{ccccccccccccc}
7 & 7 & 7 & 4 & 3 & 1 & 0 & 7 & 6 & 4 & 1 & 1 & 7 \\
 & 2 & 1 & 3 & 4 & 1 & 7 & 5 & 6 & 7 & 4 & 5 & 0 \\
 & 0 & 0 & 4 & 4 & 0 & 4 & 7 & 1 & 1 & 4 & 6 & 1 \\
 & 1 & 6 & 0 & 3 & 4 & 5 & 4 & 0 & 6 & 2 & 6 & 3 \\
 & 4 & 3 & 0 & 0 & 5 & 3 & 3 & 6 & & & &
\end{array}
$$

Decoding is done based on the proprietary key obtained in this example as m = 0.9298

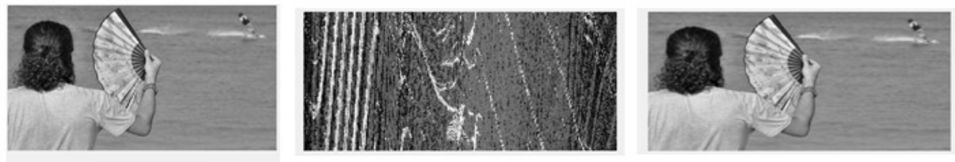
$$
\begin{array}{ccccccccccccc}
0 & 0 & 0 & 4 & 1 & 2 & 6 & 0 & 7 & 4 & 2 & 2 & 0 \\
 & 5 & 2 & 1 & 4 & 2 & 0 & 3 & 7 & 0 & 4 & 3 & 6 \\
 & 6 & 6 & 4 & 4 & 6 & 4 & 0 & 2 & 2 & 4 & 7 & 2 \\
 & 2 & 7 & 6 & 1 & 4 & 3 & 4 & 6 & 7 & 5 & 7 & 1 \\
 & 4 & 1 & 6 & 6 & 3 & 1 & 1 & 7 & & & &
\end{array}
$$

Now these codes are converted from octal to binary basis. Finally, the by coded test in the output is as follows:
'hello, i am a good student.'

Example 4.2. *(Image data example)*
In this section, the example is run for an image for encryption of the proposed method and the results are examined for encryption with the public key [0 and 1]. Based on the encryption method defined in the previous section, the encryption results of an image and finally the decoded image can be viewed. Figure 10 shows a block diagram showing this part of the program for a better understanding of the topic. The data defined in Figure 11 is shown for an image. The proprietary key obtained in this method is m = 0.0587 that varies for different images in this area. By changing the key, the coded image will also change. In this figure, image a is the same as the original image for coding. After initial processing of the image, a binary code is obtained, which is compressed and encoded by fuzzy mapping and converted to image B. Now the final image after decoding is shown in Figure 11c which is similar to the original image in quality. Figure 12 has also tested another image sample.

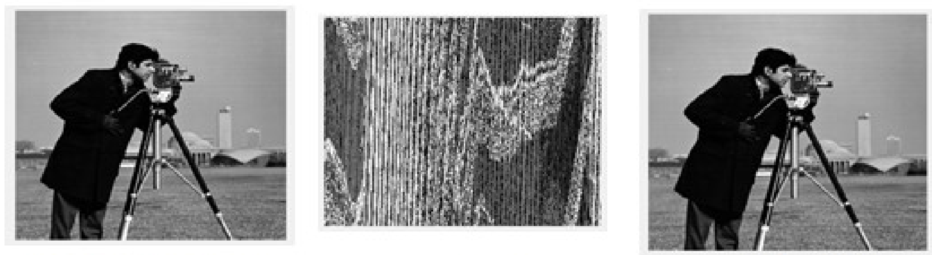
Figure 10: Display block diagram of the suggested layout for the image

As can be seen:



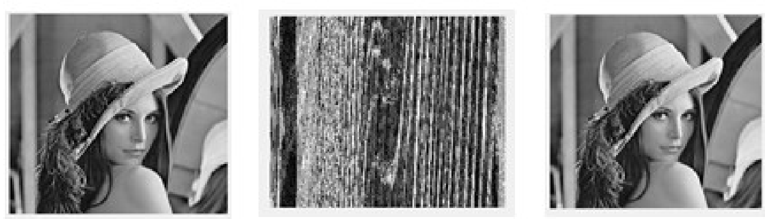
A- The original image B- Coded image C- The decoded image

Figure 11: Encryption and decryption of the image



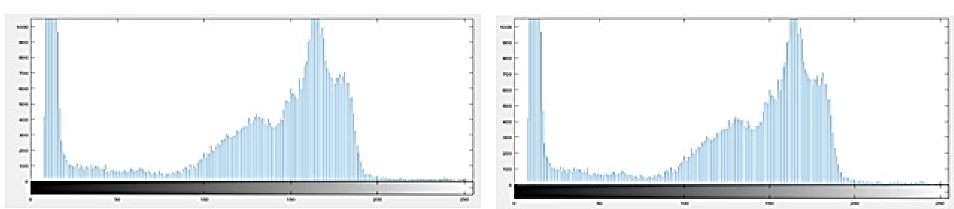
A- The original image B- Coded image C- The decoded image

Figure 12: Encryption and decryption of the image



A- The original image B- Coded image C- The decoded image

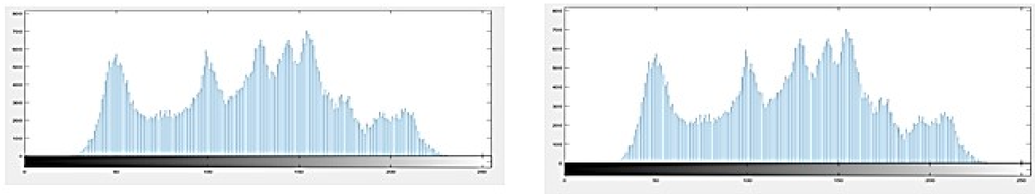
Figure 13: Encryption and decryption of the image



A B

Figure 14: Histogram representation of the cameraman image; A. Original image and B. decoded image

In this section, three image samples are used for testing. As can be seen in Figures 11, 12, and 13, the original images have changed completely after being coded and no clear images are visible. After the coded image is returned, the result of the decoding output is the same as the original image with the least amount of change without loss of information. Figures 14 and 15 show the histogram of the tested images for the original and the decoded images. Based on these figures, the complete similarity of these figures is shown under the histogram. The important point in this method is the simultaneous use of security compression and encryption of data so that image data can be compressed and stored as lossless. Moreover, in a systematic structure a security is considered for the compressed data that removes access to individuals. Table 4 shows the information size changes under this compressed method.



A B

Figure 15: Histogram representation of the Lena image; A. Original image and B. decoded image

Table 4: Volume of the coded and decoded data

Sample	Incode data(bit)	Decode data(bit)	Compression rate	[3] Compression rate
Text	216	166	1.301	-
Lena	699052	495259	1.4115	-
Black and white	699052	490000	1.4266	-
Cameraman	699052	494843	1.4126	1.32

5 Conclusion

Data security is a challenging task in IoT. Since all data is stored and maintained in the system, there is a need for data security in IoT. The proposed approach provides security and compression for data and emphasizes the use of the fuzzy Hoffman coding algorithm for compression and the use of fuzzy logic encryption for data security. The proposed method uses a fuzzy logic mechanism based on the frequency of the symbols to determine Hoffman compression weights. These symbols are derived from the classification of information codes that are moved by fuzzy logic for data security under a dedicated key. The proposed method uses symmetric key encryption. In fact, it uses a public key for encryption that based on the data structure, the number of symbols and this public key, a proprietary key that varies for each data set is achieved and security encryption and decryption is done based on this key. The proposed method has been tested under matlab2017b software for image and text information data and shows that it has been successful in data compression and security.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, *Wireless sensor networks: A survey*, Computer networks, **38**(4) (2002), 393-422.

[2] L. Fei, H. Wang, L. Chen, Y. Deng, *A new vector valued similarity measure for intuitionistic fuzzy sets based on OWA operators*, Iranian Journal of Fuzzy Systems, **16**(3) (2019), 113-126.

[3] R. K. Gangwar, M. Kumar, A. Jaiswal, R. Saxena, *Performance analysis of image compression using fuzzy logic algorithm*, Signal and Image Processing: An International Journal (SIPIJ), **5**(2) (2014), 73-80.

[4] J. Gubbi, R. Buyya, S. Marusic, M. Palaniswami, *Internet of things (IoT): A vision, architectural elements, and future directions*, Future Generation Computer Systems, **29**(7) (2013), 1645-1660.

[5] K. Hirota, H. Nobuhara, K. Kawamoto, S. I. Yoshida, *On a lossy image compression/reconstruction method based on fuzzy relational equations*, Iranian Journal of Fuzzy Systems, **1** (2004), 33-42.

[6] D. A. Huffman, *A method for the construction of minimum-redundancy codes*, Proceedings of the IRE, **40**(9) (1952), 1098-1101.

[7] A. Jafari, M. Rezvan, A. Shahbahrami, *A comparison between arithmetic and Huffman coding algorithms*, The 6th Iranian Machine Vision and Image Processing Conference, (2010), 248-254.

[8] S. Jamali, R. Fotohi, *Defending against wormhole attack in MANET using an artificial immune system*, New Review of Information Networking, **21**(2) (2016), 79-100.

[9] N. B. Karayiannis, *Applications of fuzzy logic technology II*, International Society for Optics and Photonics, **2493** (1995), 206-217.

[10] V. Kavitha, K. Easwarakumar, *Enhancing privacy in arithmetic coding*, ICGST-AIML Journal, **8** (2008), 23-28.

[11] N. H. Kumar, R. M. Patil, G. Deepak, B. M. Murthy, *A novel approach for securing data in IoTcloud using DNA cryptography and Huffman coding algorithm*, 2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS), Coimbatore, India, 2017, 1-4.

[12] G. Lakhani, *Modified JPEG huffman coding*, IEEE Transactions on Image Processing, **12**(2) (2003), 159-169.

[13] Z. N. Li, M. S. Drew, J. Liu, *Fundamentals of multimedia*, Springer International Publishing, 2004.

[14] M. Mitzenmacher, *On the hardness of finding optimal multiple preset dictionaries*, IEEE Transactions on Information Theory, **50**(7) (2004), 1536-1539.

[15] B. O'Hanen, M. Wisan, *JPEG Compression*, December 16, 2005.

[16] K. K. Parhi, T. Nishitani, *VLSI architectures for discrete wavelet transforms*, IEEE Transactions on Very Large Scale Integration (VLSI) Systems, **1**(2) (1993), 191-202.

[17] R. C. Pasco, *Source coding algorithm for fast data compression*, PHD Thesis, Stanford University CA 94305, 1976.

[18] B. Prajapati Ashishkumar, P. Barkha, *Implementation of DNA cryptography in cloud computing and using socket programming*, 2016 International Conference on Computer Communication and Informatics (ICCCI), IEEE, (2016), 1-6.

[19] Z. Qian, X. Zhang, S. Wang, *Reversible data hiding in encrypted JPEG bitstream*, IEEE Transactions on Multimedia, **16**(5) (2014), 1486-1491.

[20] M. Ramakrishnan, S. Rajkumar, *Cf-Huffman code based hybrid signcryption technique for secure data transmission in medical sensor network*, International Journal of Applied Engineering Research, **10** (2015), 11455-11474.

[21] J. Romero-Mariona, R. Hallman, M. Kline, J. Miguel, M. Major, L. Kerr, *Security in the industrial internet of things - the C-SEC approach*, International Conference on Internet of Things and Big Data, **2** (2016), 421-428.

[22] M. Sharma, *Compression using Huffman coding*, IJCSNS International Journal of Computer Science and Network Security, **10**(5) (2010), 133-141.

[23] U. K. Srivastava, *Hybrid image compression using DCT and fuzzy logic*, International Journal of Advance Research, Ideas and Innovations in Technology, **2**(6) (2016), 1-9.

[24] Y. Sudarya Triana, A. Retnowardhani, *Blowfish algorithm and Huffman compression for data security application*, IOP Conference Series: Materials Science and Engineering, **453**(1) (2018), 1-9.

[25] V. S. Thakur, K. Thakur, *Design and implementation of a highly efficient gray image compression codec using fuzzy based soft hybrid JPEG standard*, 2014 International Conference on Electronic Systems, Signal Processing and Computing Technologies, Nagpur, 2014, 484-489.

[26] C. Tharini, P. V. Ranjan, *Design of modified adaptive Huffman data compression algorithm for wireless sensor network*, Journal of Computer Science, **5**(6) (2009), 466.

[27] M. Usman, I. Ahmed, M. I. Aslam, S. Khan, U. A. Shah, *SIT: A lightweight encryption algorithm for secure internet of things*, arXiv preprint arXiv:1704.08688, (2017).

[28] M. R. Usman, M. A. Usman, S. Y. Shin, *A novel encoding-decoding scheme using Huffman coding for multimedia networks*, 2018 15th IEEE Annual Consumer Communications and Networking Conference (CCNC), Las Vegas, NV, 2018, 1-6.

[29] G. K. Wallace, *The JPEG still picture compression standard*, IEEE Transactions on Consumer Electronics, **38**(1) (1992), xviii-xxiv.

[30] R. Want, S. Dustdar, *Activating the internet of things [Guest editors introduction]*, Computer, **48**(9) (2015), 16-20.

[31] J. S. S. M. Wong, S. D. Cotofana, S. Vassiliadis, *General-purpose Huffman encoding extension*, International Conference on Information Technology: Coding and Computing. IEEE, (2000), 158-163.