Modified Lsb For Hiding Encrypted Kurdish Text Into Digital Image

Nada Elya Tawfiq
Department of Computer Science, Nawroz University, Duhok, Kurdistan Region - Iraq

ABSTRACT

Image files can hide text without their size being affected too much. This process called steganography which allows hiding text in images without any suspicions from intruders. This paper addresses an improved LSB substitution algorithm for hiding Kurdish text information written in text file into digital image as steganography technique. The algorithm consists of two main phases, the first phase holds the encryption of the Kurdish text message and the embedded technique while the second phase hold the message extraction followed by decryption to get the original code of each character. The algorithm contains many procedures to enhance this process. Least Significant Bit method is used to hide the Kurdish text, in order to keep the features and characteristics of the original image. Applying the proposed approach shows that it seems work in a best case by hiding and retrieving text from the digital image which is used as a carrier of this text. Delphi 2010 was used to simulate both encrypt-embedded phase and extract-decrypt phase, and the results were obtained with high and security which proved the efficiency of the algorithm, where the hidden Kurdish text didn't make any distortion or change over the cover image.

KEYWORDS : steganography techniques, Data Hiding, LSB substitution, Kurdish text, Delphi 2010.

1. INTRODUCTION

The major goal of steganography is to increase communication security by inserting secret message into the digital image, modifying the redundancy or nonessential pixels of the image, and is recently become important in a number of application areas especially military and intelligence agencies which require unobtrusive communications [1]. Steganography and cryptography are both ways to protect information from unwanted parties but neither technology alone is perfect and can be compromised. If the presence of hidden information is suspected or even revealed, the purpose of steganography is partly defeated. Steganography is used for transmitting data in a media such as image. Cryptography and steganography are different in their methods of hiding information. Cryptography scrambles a message and hides it in a carrier, so that if it is intercepted it would be generally impossible to decode. Steganography hides the very existence of the message in the carrier. When the message is hidden in the carrier a stego-carrier is formed e.g. a stego-image. If successful, it would be perceived to be as close to the original carrier or cover image by the human eye. Images are the most widespread carrier medium. They are used for steganography in the following way: The message may firstly be encrypted. The sender embeds the secret message to be sent into a graphic file. This results in production of what is called the stego-image. Additional secret data may be needed in hiding process e.g. a stego key. The stego-image is then transmitted to the recipient. The recipient or extractor extracts the message from the carrier image. The message can only be extracted if there is a shared secret between the sender and the recipient. This could be the algorithm for extraction or a special parameter such as a key (the stego-key). To make the steganographic process even more secure the message may be compressed and encrypted before it is hidden in the carrier.

Academic Journal of Nawroz University
(AJNU) Volume 7, No 4 (2018).
Regular research paper : Published 5 Jan 2019
Corresponding author's e-mail : nada.elia@yahoo.com

2. WHAT IS STEGANOGRAPHY

In modern times, steganography can be looked into as the study of the art and science of communicating in a way, which hides the existence of the communication, has until recently been the poor cousin of cryptography. One of the most common uses of modern steganography in the digital world of computers is to hide information from one file in the contents of another file. [3] Figure (1) illustrates the principles of steganography where a carrier image has a message is added and put through a Stegosystem Encoder. The stego-image then will be sent through the appropriate channels to Stego-system Decoder as shown in figure (1). [2]

doi : 10.25007/ajnu.v7n4a298

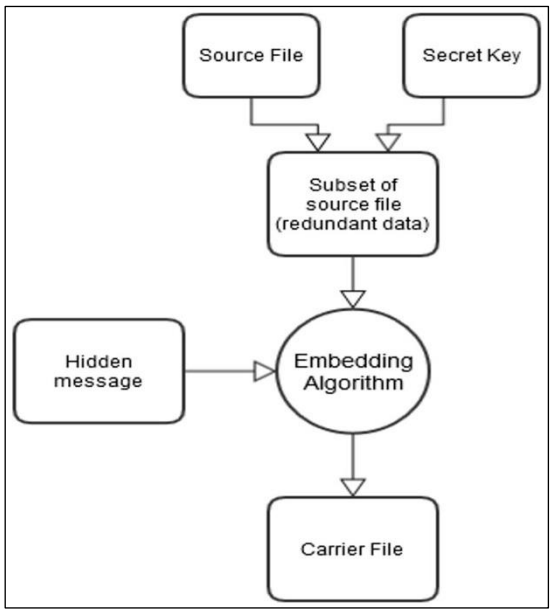

Fig (1) : General steganographic mechanism

When the messages are transmitted over an untrusted channel, it means that they need to be protected, mainly there are two scenarios for this purpose [4] :

1. Cryptography : Hidden data meaning.
2. Steganography : Hidden data existence.

Usually the first scenario (cryptography) was used where data transferred from readable form (plain text) into scribbled data (cipher text), and then rely on transmitting cipher text message by using a secrete key. While with the second scenario, steganography will hide data existence by using another file as carrier and a strategy to insert secret data inside it. Then will pass the data through the communication channel. In this case everyone can read the carrier file but no one can notice the hidden message. Steganography imaging system is a system that capable of hiding the data inside the image. The system is using 2 layers of security in order to maintain data privacy. Data security is the practice of keeping data protected from corruption and unauthorized access [5].

2.1 HIDING DATA IN IMAGES

The purpose of steganography is to avoid drawing suspicion to the transmission of hidden information. A message is a hidden information in the form of plain text, cipher text, images or anything that can be encoded into a bit stream. This message is embedded in a cover-carrier to create a stego-carrier. A possible formula of the process may be represented as follows [6] :

Stegomedium = Covermedium + Embeddedmessage + Stegokey Turn MathJaxon Choosing carrier file is very sensitive as it plays a key role to protect the embedded message.

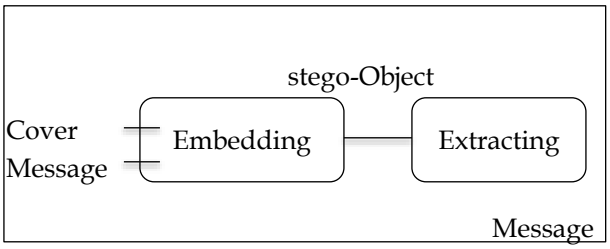

Fig (2) : Embedded & Extract Data

2.2 STEGANOGRAPHY TECHNIQUES

The secret message is embedded inside the cover object in encrypted format by using a hiding algorithm and it be sent to a receiver over a network. The receiver then decrypts the message by applying the reverse process on the cover data and reveals the secret data. [7] The steganography system has two inputs as shown in Figure (3), a "Cover Object" and a "Secret Object", which is confidential. Steganography algorithm comes into picture to do the embedding part for the two inputs, i.e. these two objects output the "Stego Object" [7].

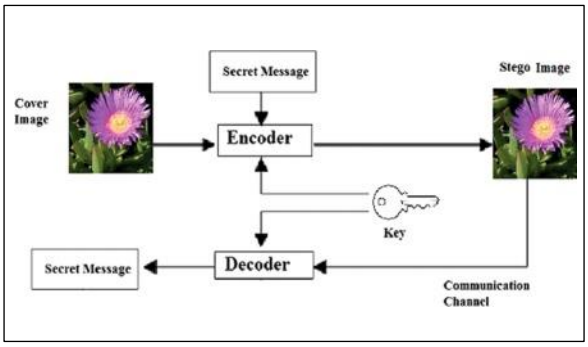

Fig (3) : Basic Steganography Mode

There are many techniques for hiding information or messages in images, common approaches are including [6] :

1. Least significant bit insertion (LSB).
2. Masking and filtering.
3. Transform techniques.

2.3 STEGANOGRAPHY IMPLEMENTED USING LSB

There are several different techniques for concealing data inside of normal files. One of the most widely used and perhaps simplest to understand is the least significant bit technique, known commonly as LSB [8].

This technique changes the last few bits in a byte to encode a message, which is especially useful in something like an image, where the red, green, and blue values of each pixel are represented by eight bits (one byte) ranging from 0 to 255 in decimal or 00000000 to 11111111 in binary. Changing the last two bits in a completely red pixel from 11111111 to 11111101 only changes the red value from 255 to 253, which to the naked eye creates a nearly imperceptible change in

doi : 10.25007/ajnu.v7n4a298

color but still allows us to encode data inside of the picture. [8] Two other things taken in consideration are encryption and compression. Encrypting data before embedding it adds an extra layer of security while compressing your data will obviously allow you to fit more into your cover file. Both encryption and compressions schemes can be included as optional parameters in Steg-hide [8].

2.4 DATA HIDING BY LSB SUBSTITUTION

Least Significant Bits (LSB) insertion is a simple approach to embed data in image file. The easiest steganographic methods embed the bits of the message directly into least significant bit plane of the cover-image in a sequence. Modulating the LSB does not camp up with in human-perceptible difference because the amplitude of the change is small [9].

3. STRUCTURAL FEATURES OF KURDISH ALPHABETS

The Kurdish languages are written in either of two alphabets : a Latin alphabet introduced by Jeladet Ali Bedirkhan (Celadet Alî Bedirxan) in 1932 Bedirxan alphabet, or Hawar after the Hawar magazine), and a Persian alphabet-based Sorani alphabet, named for the historical Soran Emirate of present-day Iraqi Kurdistan. The Kurdistan Regional Government (KRG) has agreed upon a standard for Sorani, implemented in Unicode for computation purposes. The Hawar is used in Turkey, Syria and Armenia; the Sorani in Iraq and Iran [10].

3.1 HAWAR ALPHABET

Kurmanji dialect of the Kurdish language is written in an extended Latin alphabet, consisting of the 26 letters of the ISO basic Latin Alphabet with 5 letters with diacritics, for a total of 31 letters (each having an uppercase and a lowercase form) as shown in Table (1). In this alphabet the short vowels are E, I and U while the long vowels are A, Ê, Î, O and Û (see the IPA equivalents in the Help:IPA/Kurdish table). When presenting the alphabet in his magazine Hawar, Jeladet Ali Bedirkhan proposed using ⟨ḧ ẍ ˈ⟩ for غ, ح, and ع, sounds which he judged to be "non-Kurdish". These three glyphs do not have the official status of letters, but serve to represent these sounds when they are indispensable to comprehension. [11].

TABLE (1) : Hawar alphabet

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24		25	26	27	28	29	30	31
	Majuscule forms (also called uppercase or capital letters)																														
A	B	C	Ç	D	E	Ê	F	G	H	I	Î	J	K	L	M	N	O	P	Q	R	S	Ş	T		U	Û	V	W	X	Y	Z
	Minuscule forms (also called lowercase or small letters)																														
a	b	C	Ç	d	e	ê	f	g	h	i	Î	j	K	l	m	n	O	p	q	r	s	ş	t		u	û	v	w	x	y	z

consonant "w" from the short vowel "u" by representing "w" with a [ڤ]. It is also able to successfully differentiate between the consonant "y" from the long vowel "î" by representing "î" with a [ێ] and the long vowel "û" can be represented with a [ۆ] or [ۇ] instead of double و. Figure (4) illustrates the Sorani Alphabet [10] :

doi : 10.25007/ajnu.v7n4a298

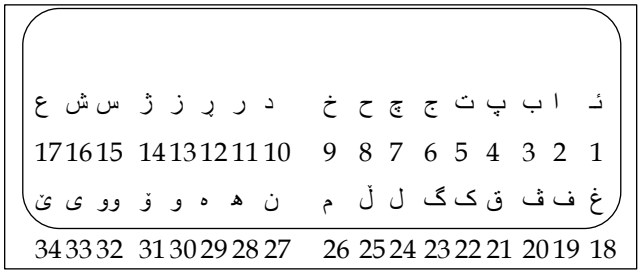

Fig (4) : Sorani Alphabet

Note - The above sequences are read from right to left.
A new sort order for the alphabet was recently proposed by the Kurdish Academy as the new standard, all of which are letters accepted included in the Sorani Unicode Keyboard.

3.3 FORMS OF KURDISH ALPHABETS

Kurdish characters can have more than one shape according to their position in a word whether it is in the beginning, middle, final, or standalone. Table (2) contains an example of different forms for Kurdish letter with the International Phonetic Alphabet (IPA):[10]

TABLE (2) : different forms for Kurdish letter

Isolated	final	medial	initial	IPA
ا	ـا	ـا	ئا	[aː]
ب	ـب	ـبـ	بـ	[b]
ج	ـج	ـجـ	جـ	[dʒ]
چ	ـچ	ـچـ	چـ	[tʃ]
د	ـد	ـد	د	[d]
ه	ـه	ـهـ	ئه	[ɛ]
ێ	ـێ	ـێـ	ئێـ	[eː]
ف	ـف	ـفـ	فـ	[f]
گ	ـگ	ـگـ	گـ	[g]
ھ	—	ـھـ	ھـ	[h]
ی	ـی	ـیـ	—	N/A
ی	ـی	ـیـ	ئیـ	[iː]
ژ	ـژ	ـژـ	ژـ	[ʒ]
ک	ـک	ـکـ	کـ	[k]
ل	ـل	ـلـ	لـ	[l]
ڵ	ـڵ	ـڵـ	—	[ɬ]
م	ـم	ـمـ	مـ	[m]
ن	ـن	ـنـ	نـ	[n]
—	—	—	—	[ŋ]
ۆ	ـۆ	ـۆـ	ئۆ	[o]
پ	ـپ	ـپـ	پـ	[p]

The alphabet is represented by (34) letters including (وو) which is given its own position. Kurds in Iraq and Iran mainly use this alphabet, though the Kurdish Latin alphabet is also in use. [10]. However, the latter glyph is still in use by various individuals and organizations. Table (3) shows the Unicode Kurdish characters :

TABLE (3) : Unicode Kurdish characters

Kurdish	Unicode name (Kurdish letters)	Unicode
ئ	Yeh with Hamza above	0626
ا	Alef	0627
ب	Beh	0628
پ	Peh	067E
ت	The	062A
ج	Jeem	062C
چ	Tcheh	0686
ح	Hah	062D
خ	Khah	062E
د	Dal	062F
ر	Reh	0631
ڕ	Reh with small V below	0695
ز	Zain	0632
ژ	Jeh	0698
س	Seen	0633
ش	Sheen	0634
ع	Ain	0639
غ	Ghain	063A
ف	Feh	0641
ڤ	Veh	06A4
ق	Qaf	0642
ک	Keheh	06A9
گ	Gaf	06AF
ل	Lam	0644
ڵ	Lam with small V	06B5
م	Meem	0645
ن	Noon	0646
و	Waw	0648
ۆ	Oe	06C6
وو	U	06C7
ھ	Heh Doachashmee	06BE
ه	Heh	0647
ی	Farsi Yeh	06CC
ێ	Yeh with small V	06CE

4. PROPOSED ALGORITHM

The proposed technique contains algorithms of encryption and embedding data to embed the cipher text in a cover medium. The system combines effect of these two methods to enhance the data security. First method encrypts the data with a crypt algorithm, then embeds the encrypted data in a cover file. The block diagram of the proposed system is shown in figure (5).

doi : 10.25007/ajnu.v7n4a298

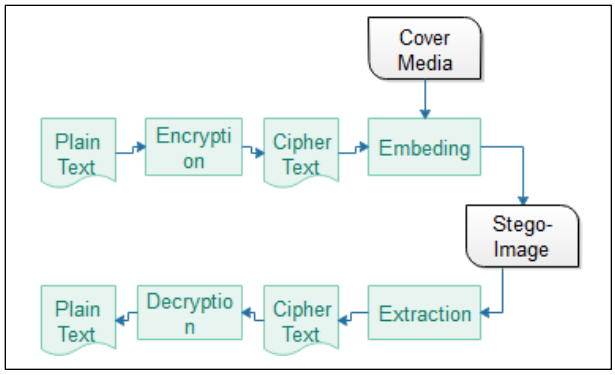

Fig (5) : steps of proposed algorithm

Least Significant Bit (LSB) technique in the embedding stage is used, which is most commonly used due to its simplicity and easy implementation. The change in the LSB of the pixel in image renders the visual quality of image on hiding the message does not much affect the image and make it unnoticeable to unintended person. The steg_ image in proposed algorithm refers to crypted text embedded with the imge. This steg_ image will be In order to hide the message produced after encryption, a new method was proposed to embed the message which contains Kurdish text within the image as the cover media. The result would produce a steg_image which then be transmitted over the channel. In this section, the colored BMP images are used as cover medium and the procedure is addressed. The Unicode of Kurdish characters is beginning from 0626 as shown in table (4) which includes all characters in different forms. The proposed technique consists of two major stages involved in data hiding and extraction:

Stage 1 : - Hiding Algorithm :

1. Input BMP-image.
2. Convert each image pixel value from decimal system into binary digital system.
3. Input the text message, and then convert each character into binary number.
4. Encrypt the text message using secret key then, the bits of each character will be embedded in bits LSB part of image pixel.
5. Output stego-image.

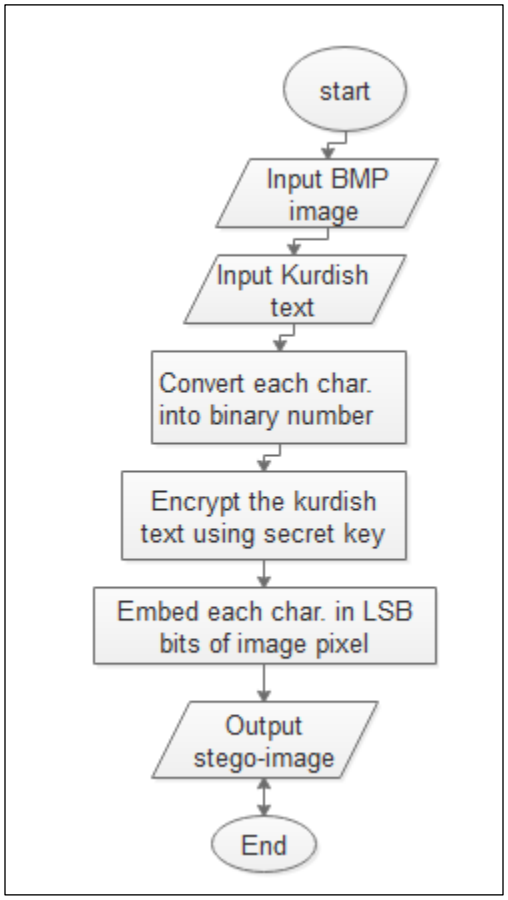

Fig (6): Flowchart of Hiding algorithm

Stage 2 : - Extract hiding information Algorithm:

1. Input stego-image.
2. Convert each image element into binary system.
3. Extract from each image element first 2 bits (LSB). Then arrange these data into a stream of bits.
4. Divide the stream of bits into blocks each block contains 16 bits.
5. Then each 16 bits is decrypted using the secret Key and converted into represented symbols, to reconstruct the text-message again.
6. The output is the text message.
7. End algorithm

doi : 10.25007/ajnu.v7n4a298

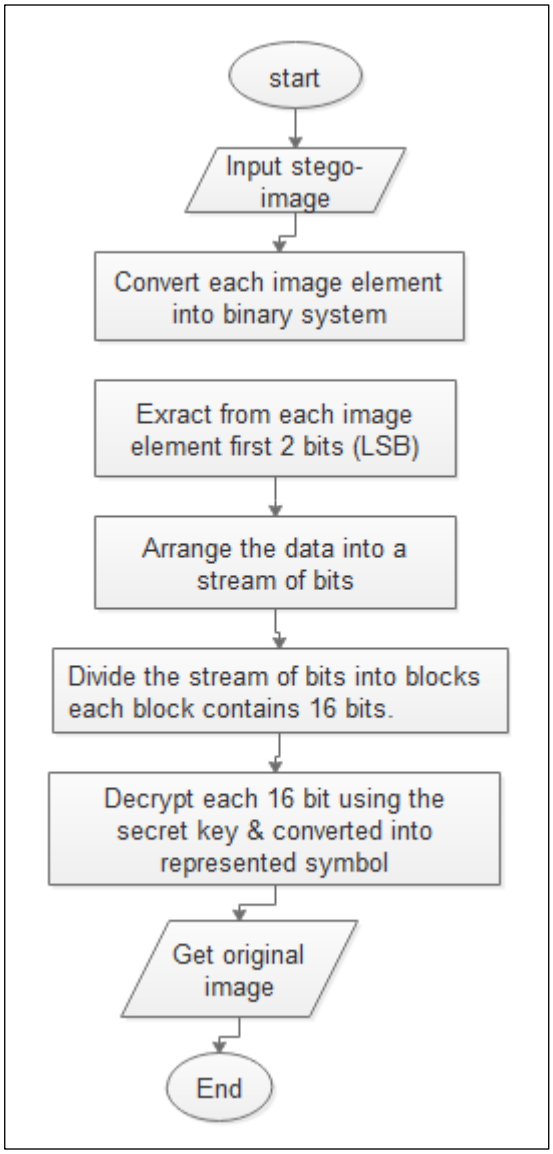

Fig (7) : Flowchart of Extract algorithm

4.1 METHODOLOGY

The first step in the proposed algorithm is to converting data into bytes in which each character in the Kurdish text is converted to its equivalent Unicode.

For example :

The Unicode of character (چ) is 0686 and equivalent binary is (0000001010101110) plus the code of the key (101) = (0000001010110011).

For example, RGB & alpha are : - 00100111- 11101001– 11001000 – 00101001

Then the message is embedding 2-bits into the LSB position of each pixel position in RGB, as shown below:

The RGB values will be: - 00100110 - 11101011 – 11001000 - 00101011

We can save any number of characters into 24-bit BMP image.

Fig (8) : represents example of images embedded with Kurdish text

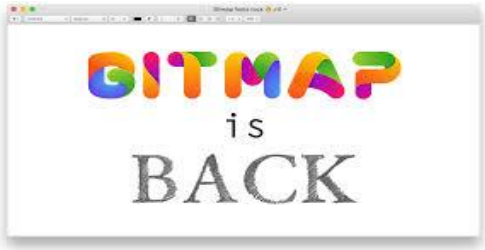

Image1 before embedding text

Image1 after embedding the text

Image2 before embedding

Image2 after embedding

Fig (8) : Images before & after embedding Kurdish text

The next step is embedding Kurdish text within selected image, then saving the new image that contains hidden text. Finally extracting the text from the image.

doi : 10.25007/ajnu.v7n4a298

5. RESULTS

The results obtained by this algorithm indicate that the embedding process introduces high Peak Signal to Noise Ratio (PSNR) with less perceptual distortion. This is meant that the quality degradations could hardly be perceived by a human eye. The PSNR (peak signal-to-noise ratio between two images) & MSE (Mean Square Error, the cumulative squared error between the compressed and the original image) for the proposed algorithm when applied on many images is tabulated in table (4) as shown below :

TABLE (4) : PSNR & MSE for test images

Stego-image	PSNR	MSE
Image1	67.6164	3.7559
Image2	72.4925	0.0037
Image3	64.8610	0.0212
Image4	66.7610	0.0145

Mean Square Error:

$$MSE = \frac{1}{m*n} * \sum_{i=1}^{m} \sum_{j=1}^{n} (CI(i,j) - SI(i,j))^2 \ldots . . (1)$$

Where : CI is cover-image of size (M x N)
SI is stego-image (M x N), $1 \le i \le M$,
$1 \le j \le N$.

The equation to calculate Peak Signal to Noise Ratio (PSNR) :

$$PSNR = 10*\log_{10}\left(\frac{R^2}{MSE}\right) \ldots \ldots \ldots \ldots \ldots \ldots . . (2)$$

If cover-image is gray scale image of integer values [0-255], then R=255.

6. CONCLUSION

The practical results have proved that the suggested algorithm is sufficient in terms of no much changes nor noticeable distortion appeared at the hidden information on the used cover file. Many other features can be noticed such as:

1. With regard to the results of PSNR for different file sizes which demonstrated in table (4), the values of the PSNR are sophisticated and really very good, that is because when PSNR is high, it is better.
2. The system has been built to provide high security in a style to be very difficult to infer and detect the existence of any hidden information.
3. It is possible to hide huge amount of data in the existed image.
4. The hiding system can be used with high flexibility and simplicity.
5. with PSNR measure a large number implies a better stego-imag.

7. FUTURE WORK

As a future work, it is suggested to design an algorithm to hide text in other languages and use more conflicted key, as well as the image can be compressed.

REFERENCES

1. Deepesh Rawat MTech , Vijaya Bhandari, April (2013), "Steganography Technique for Hiding Text Information in Color Image using Improved LSB Method", International Journal of Computer Applications (0975 – 8887) Volume 67– No.1, INDIA.
2. Dr. Salman Abd Kadum, Tameem Hameed Abaidah, (2010), "Enhancement an Algorithm to Hide a text into a Digital Image as a Steganography Technique"
https://www.iasj.net/iasj?func=fulltext&aId=48620/
3. Kefa Rabah , (2004), "Steganography-The Art of Hiding Data", Information Technology Journal ,Volume 3 (3): 245-269.
4. Nada Elya Tawfiq, baraa salim, (2015), "An enhanced steganography technique for crypting&hiding arabic text in to digital image", Journal of university of Duhok, Volume 3, No. (1).
https://www.uod.ac/
5. Ammar Odeh, Khaled Elleithy, Miad Faezipour, (2013), "Steganography in Arabic Text Using Kashida Variation Algorithm",
http://www.khaledelleithy.org/Conferences/06578239.pdf
6. https://www.null-byte.wonderhowto.com/how-to/steganography-hide-secret-data-inside-image-audio-file-seconds-0180936/
7. Mr. Falesh M. Shelke1, Miss. Ashwini A. Dongre2, Mr. Pravin D. Soni3, February (2014), "Comparison of different techniques for Steganography in images ", International Journal of Application or Innovation in Engineering & Management (IJAIEM) Web Site: www.ijaiem.org Email: editor@ijaiem.org, editorijaiem@gmail.com Volume 3, Issue 2, ISSN 2319 - 4847].
8. https://null-byte.wonderhowto.com/how-to/steganography-hide-secret-data-inside-image-audio-file-seconds-0180936/
9. Anwar H. Ibrahim, Waleed M. Ibrahim, January / February, (2013), " Text Hidden in Picture Using Steganography: Algorithms and Implications for Phase Embedding and Extraction Time", International Journal of Information Technology & Computer Science (IJITCS) (ISSN No : 2091-1610) Volume 7 : No : 3.
10. Cabinet.gov.krd , Retrieved 2016-03-01, "Kurdistan Regional Government (Kurdish article)".
11. Kyumars Sheykh Esmaili, (2013), "Sorani Kurdish versus Kurmanji Kurdish: An Empirical Comparison" Conference paper.
https://www.researchgate.net/puplication/270877570