Journal of Electrical Engineering, Electronics, Control and Computer Science
JEEECCS, Volume 3, Issue 8, pages 7-20, 2017

SMS Blind Steganography by Using Maximum-Minimum Method

Abouzar Zare
Department of Computer (Computer Engineering) Dariun Branch, Islamic Azad University, Dariun, Shiraz, Iran
arzare.ir@gmail.com

***Abstract* – This paper represents an effective steganography method for the textual data in black and white images used in the pictorial short message service (SMS). Practically, compare with other methods, the steganography capacity increases substantially by changing the selection methodology and the style of blocking as well as the way of storing characters. In order to have bit changes in a block of the cover image, such that the change would not be noticeable by an observer (to be imperceptible), maximum of one-bit change per block is allowed. The introduced maximum-minimum method is based how the required pixel value, which is assumed to be located in the center of the block, is related to the number of the black or white pixels in that block. In fact, only the center pixel value maybe changed in a block. The hidden secret message may be recovered with examining a pixel with its related block; and so, there is no need to have the original image. By comparing it with other methods, our new method offers higher capacity, more efficient imperceptibility, and better security features.**

Keywords- Information Hiding, Steganography, Short Message Service (SMS)

I. INTRODUCTION

Today, digital media are widely used worldwide. One of most popular digital media is the cell or mobile phone. This tool has changed human life extensively to a level no one could imagine before. It is offering new facilities every day and it upgrades human ability drastically. The first short message sent in the world was occurred in 1992. The use of this service was drastically expanded in the last two decades such that about 80 percent of the people are using this service in frequent bases now. Along with all the advantages it offers, it also brings new challenges. Among them it is illegal access of private conversation in its phone feature and confidential data in its short message service (SMS) feature by intruders.

The length of a SMS is limited and it varies with the coding scheme. The popular coding standards that are used in SMS and the text length per message segment in each standard are presented in Table.1. For example, the table shows that the GSM 7-bit default alphabet - which is defined in 3GPP-23.038 standard - consists of 128 characters plus 9 additional characters (extension table) including the Euro signs. Another example, the Universal Character Set with 2-octet symbols (UCS2) – which is defined in ISO-10646 standard and is utilized for encoding various sets of non-Latin characters such as Chinese and Arabic – consists of 70 complex characters. To see which characters are available in each coding scheme, one can refer to reference [9].

Table.1: The relation between the coding scheme and the length of the text.

Coding scheme	Text length per message segment	Confirmed Standard
GSM alphabet, 7 bits	160 characters	3GPP-23.038
8-bit data	140 octets	GSM 03.40
UCS2, 16 bits	70 complex characters	ISO-10646

One of the important issues in today's SMS communication, especially in exchanging SMS in mobile phones, is its security. For instance, safe transmission of the subscriber's banking information bares a great significance. That is, protecting the customers' accounting information becomes a very important issue. One of effective solutions may be SMS steganography.

Note that, characters are the combination of letters, numbers and special signs, which eventually are stored as binary form. In addition, the black and white (B&W) images may also be exchanged in a SMS system, known as pictorial SMS. Similar to characters, the structure of B&W images used in SMS is a set of zeros and ones, form a binary structure. Hiding crucial information, such as accounting information, using steganography in B&W images could be an effective complement solution (along with other security measures) for increasing SMS data exchange security.

Steganography is a technique in information hidden communication for embedding confidential data in a proper cover media such as textual, vocal and pictorial files [5].

The main purpose in steganography is hiding data in another media in the way that the existence of the data would not be noticeable by an observer. The priority of this method to the other methods of hiding data like cryptography is in the same subject. In cryptography, the original message will be encrypted as a new message; so that the others can see the encrypted message.

Table.2: Comparison of steganography and cryptography

Criterion/ method	Steganography	cryptography
Carrier	Any digital media	Usually text based, with some extensions to image files
Secret data	Payload	Plain text
Key	Optional	Necessary
Input files	At least two unless in self-embedding	One
Detection	Blind	Blind
Authentication	Full retrieval of data	Full retrieval of data
Objective	Secrete communication	Data protection
Result	Stego-file	Cipher-text
Concern	Delectability/ capacity	Robustness
Type of attacks	Steganalysis	Cryptanalysis
Visibility	Never	Always
Fails when	It is detected	De-ciphered
Relation to cover	Not necessarily related to the cover. The message is more important than the cover	N/A
Flexibility	Free to choose any suitable cover	N/A
History	Very ancient except its digital version	Modern era

Therefore, in the other words, cryptography emphasize on the message while steganography is considered to hide the connection. A brief review of the comparison between three methods of data hiding (steganography and cryptography) is given in table2.

Basically, in data hiding, three considerable characters are capacity, robustness and security. These three features are closely interconnected, e.g. by increasing the robustness, the hiding capacity will be decreased and vice versa. In steganography, the features of capacity and security are very important, so that the noticeable and the perceptible (discoverable) changes won't occur in cover media.

Steganography methods are consisted of two embedding and extracting algorithm. A sender to embed data inside a media needs two significant files:

1. Cover media: a media on which the data will be hidden such as textual, vocal, and pictorial files and so forth.

2. Secret message: a message which be hidden in the media.

Usually for a better security, the secret message before being embedded in a media will be encrypted by a secret key (share between sender and receiver). A new media is produced by embedding secret message into cover media which is called stego media. After the data extracting from stego media the receiver can reach to the main data by using the key. The steganography process is featured in figure1.

The mathematical mode of steganography process is defined to formulas (1) and (2).

$$Emb : C \oplus K \oplus M \rightarrow C' \qquad (1)$$

$$Ext : C' \rightarrow M \quad \forall c \in C, \quad k \in K, \quad and \quad m \in M \qquad (2)$$

C is considered (denoted) as the indicator of cover media, k as optional secret key, M for secret message and C' for stego media; Emb and Ext are a relatively referring to embedding and extracting algorithm.

Basically, steganography algorithms are done in spatial and transform domains. In each of these domains the data would be hidden in different ways. The spatial domain is consisted of algorithms which the message bites are embedded within the media. But In transform domain at first, the media is transform to another domain such as frequency domain, then the data bites are distributed on the whole or some parts of the host bites.

The simplest way in spatial domain is using of the least significant bits (LSBs); several other methods in this domain are given in [6, 7]. A practical example of embedding in the 1st LSB and up to the 4th LSB is illustrated in figure2.

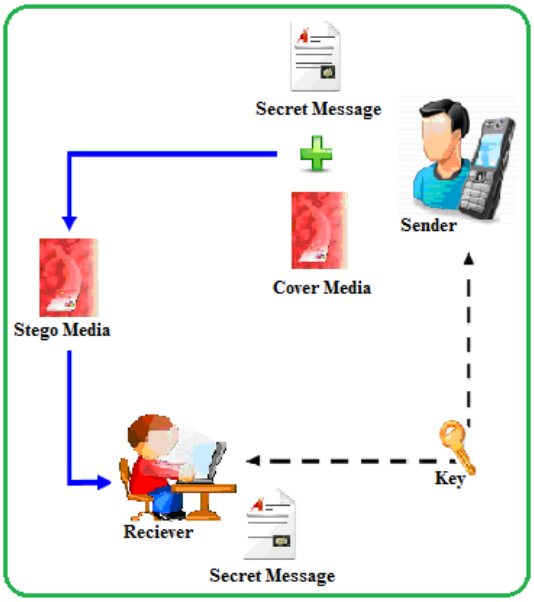


Figure1. The process of steganography

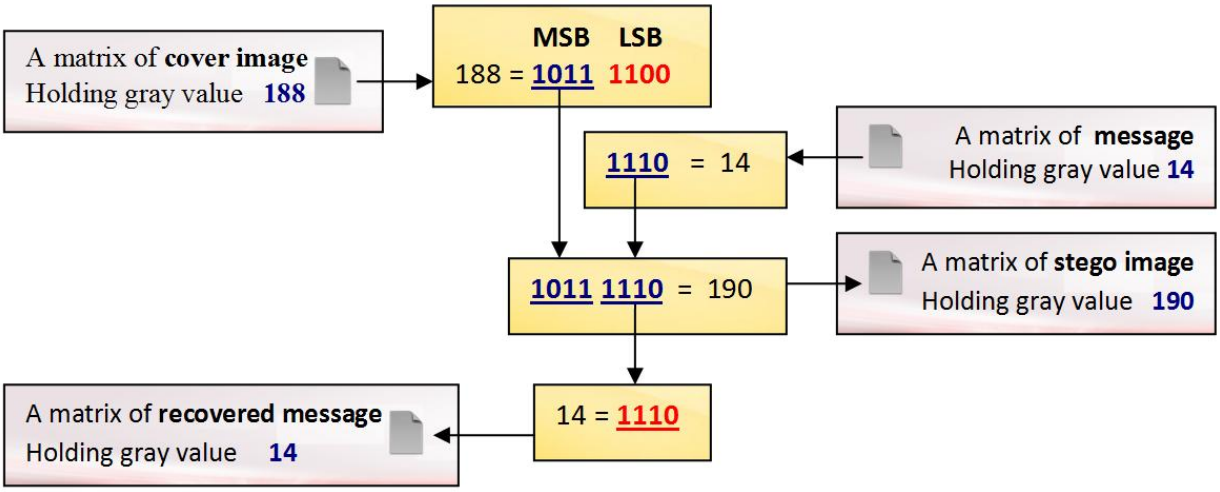


Figure2. Steganography by using of LSBs method

In transform domain, the transforms coefficients of DCT and DFT can be mentioned; further methods of this domain are indicated in [1, 5, 6, 7]. Most of researches in steganography are done on images and among all images, the researches are often done on color and grayscale images; but there have not been lots of work on B&W images due to their sensitivity and perceptible to change. Choosing a suitable image format has an effective potential on steganography capacity, and also on hidden communications discovery by the intruder. The most popular image formats used on the internet are BMP, PNG, JPEG, GIF, etc and in B&W images of mobile phones are WBMP, OTA, OTB, etc. Some uncompressed formats like bitmap images provide a high capacity for steganography. The size of secret message is interrelated to steganography capacity of image. In the other words, the image should be capable to embed the secret message.

By considering to widely use of mobile phones and transferring a great quantity of SMSs and also the ability of sending B&W image through this service, as well as advantages of this service, a proper option for secure connections, is data hiding on B&W images of SMSs [10].

The main concentrate in this paper is representation a practical method for steganography in B&W images in mobile phone SMSs; so that a safe way is make to send secret information by this service.

In this Article Firs of all, in section 2 is investigated pervious works about B&W images and SMS, afterwards, in section 3 is explained proposed method, then comparison and evaluation of proposed method is given in section 4, and finally section 5 shows the final conclusion, elaborated.

II. PERVIOUS WORKS

This section attempts to give an overview of the most important steganographic methods in B&W images. The B&W images will be tangible and sensitive to changes. Because, changing a pixel of image either in B&W image area will be quite visible. For this reason, researches done on steganography in B&W images are very few. However, this section is explained some of performed works about steganography in B&W images and SMSs.

RAFAT in [22] by creating a new SMS language transfers secret information with SMS. Due to size limitation of SMS, the steganography capacity of this method is very few.

Shir-Ali in [4] has proposed a new method by applying changes and displaces in image appearance. There should be a great bank of images to hide a numerous number of data.

Zhi-Hui in [35] has submitted a method of text steganography by embedding data in moving icons, which are used in chatting.

Shunquan Tan et.al. in [37] propose the pixel-decimation-assisted steganalytic feature set, a novel feature set construction protocol that extends upon the recent selection-channel-aware spatial rich model maxSRMd2. This method is based on pixel decimation, a specific type of image down sampling.

2.1. The change of n bits in each block of the image

The image, in this method, is first divided into M×N blocks. Afterwards each block will be changed by applying the XOR function, and encodes with using a key. Of course, regarding the weight of matrix in each block, maximum *n* bits of each block is changeable, so that each M×N block of image hides $n\log_2^{[(M\times N)/n+1]}$ bits of data [13]. The similar method is presented in [17, 18, 19], so that for each M×N block to embed $\log_2^{(M\times N+1)}$ bits by changing at most 2 bits in it. The advantage of these two methods is high steganography capacity, and their principal defect is visibility of changes on the output image.

2.2. The change in a bite of each block of image

In this method, the binary image is divided into M×N blocks, then possibility of changing a pixel

estimates in each block; and a score is given to each block. If the score exceeds a definite size, that block at most can save one bite in itself. The begotten changes in output image are not tangible due to the steganography capacity is very limit (bound-narrow) [12]. A similar method is proposed for data steganography in JPEG images by Ajetra et.al. [32], too.

2.3. Stealth Steganography

Shir-Ali[11] is proposed the method similar to section 2.2. But in this method for embedding a bit in each block is used odd-even method. In odd-even algorithm, the white pixels of each block are first counted, and by exchange middle pixel in the block, the number of white pixels changes to odd or even. For storing a bit with 1 value, the number of block white pixels should be even; and to store a bit with 0 value, the number of block white pixels should be odd. The disadvantage of this method is the low steganography capacity.

2.4. The Block Parity

Venkatesan et.al. [24], in another method, by using the parity feature in blocks, hide bits in them. The image is first divided to M×N blocks. One bit only can be embedded into each block. Then, a neighbor matrix will be produced. That each value of matrix is obtained adjacent bits number in the block which similar values have with that bit. The bit with more adjacent will be picked for embedding. With implementation odd-even feature in its adjacent bit, a bit store in the block. The bits which get hidden in each block are located in different places so that the real data are not easy discoverable in stego image; but the steganography capacity is very few [24].

2.5. New Odd-Even Method

In this method, at first B&W images are divided to 3*3 blocks. At most a bit can be hidden in each block. By changing in odd-even method and style of blocking, this method could be increased steganography capacity. In the new odd-even method, the number of white pixels on the main diagonal should be counted figure3.

To hide a bit with zero value, the white pixels number on the main diagonal must be odd, so that if they are even, by flipping the middle-pixel, they will be odd, Figure3-(a). For hiding a bit with one value, the white pixels number on the main diagonal must be even Figure3-(b); so that, if they are odd, by reversing the middle-pixel they will be even [3].

The changes in this method are few and intangible, because of some of bits are stored in some blocks without flipping the mid-pixel Figure3-(b). In section 4, is illustrated that this method is better than Shir-Ali method [11] (Stealth Steganography).

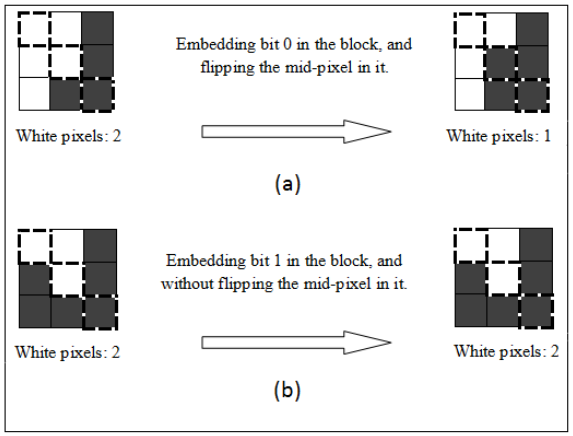


Figure3. Hiding a bit in a 3×3 block by New Odd-Even method

2.6. Sudoku Puzzle

In this method, secret data is hidden in sudoku puzzle and sent via SMS. Steganography algorithm is based on embedding 1 to 9 numbers into a special column and row after solving the puzzle. After hiding data in sudoku puzzle, the puzzle will be sent intangibly. Data extraction in this algorithm after solving the puzzle is done based on the order of 1 to 9 numbers equivalent to hidden data, in a special column and row. The main disadvantage of this method is the low capacity for data steganography [4].

In all above-mentioned methods, the steganography capacity is limited, so that these only few number of characters can be hidden in each B&W image. In this article, the steganography capacity increases substantially more than the previous similar given methods, by changing the selection methodology and the style of blocking as well as the way of storing characters.

III. PROPOSED ALGORITHM

In this section is brought the summery of proposed methods and implemented algorithm for data steganography in B&W images in SMS. In binary images, each bit is stored in one pixel. The white pixels are indicated zero bits and the black pixels are indicated one bits (the black color is the indicator of zero bit and the white color of one bit).

Some of the most functional and popular formats of binary images are: WBMP, OTA and OTB. The size of binary images for each SMS should be 72×28

Table.3: The bytes of header file in mobile phone's binary images.

Image Binary Type	***OTA and OTB***		***WBMP***	
Byte Number	***Binary Value***	***Decimal Value***	***Binary Value***	***Decimal Value***
1	0000 0000	0	0000 0000	0
2	0100 1000	72	0000 0000	0
3	0001 1100	28	0100 1000	72
4	0000 0001	1	0001 1100	28

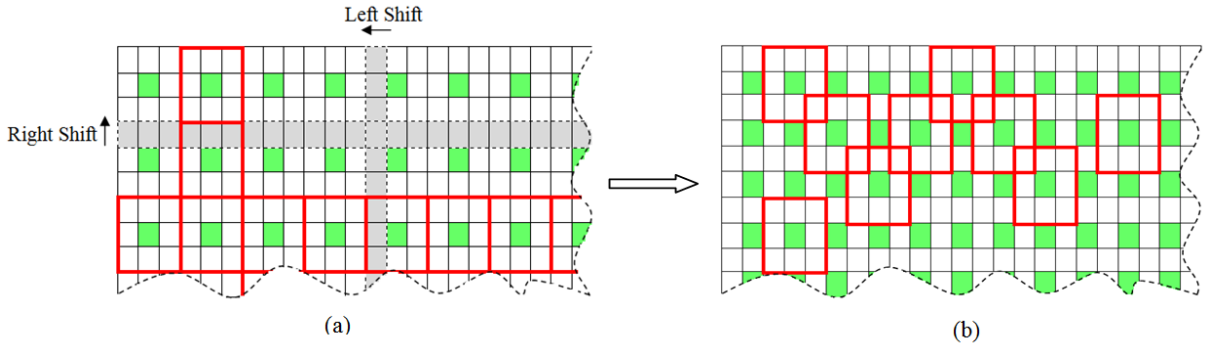


Figure5. (a) division of image to 3×3 blocks in resource [12] and suggestion shifting in blocks, (b) new style of blocking in image. The determination pixels may be exchange its color for embedding a bit in its block.

pixels. To see structures of these formats, refer to resource [13]. The size of each picture message in SMS is 256 bytes, Relation (3).

$$\left[(72\times 28)\div 8\right]+4=256 \quad byte \qquad (3)$$

There are four extra bytes which are related to header file of image which is (they are) keeps data definition and the size of binary images. The content of each of these four bytes is shown in table (3).

A way for data embedding in binary images is selecting some pixels of the image and flipping their color to store data bits into the image. Considering to changing of the image must be intangible, so, choosing the proper pixels is a fundamental factor. Also, steganography capacity is very important. For reach to these goals, at first in 5.1 part, the style of image blocking is elaborated for selecting and changing suitable bits in B&W images, then in 5.2 part, the new MAX-MIN method is proposed for embedding a bit in each block. After, it is explained in part 5.3, the way of coding characters in image as well as the summery of Implemented algorithm for steganography in B&W images on SMS.

3.1. Blocking and Flip-ability Algorithm

The marginal image pixels are the best options for data embedding in B&W images. Human Visual System (HVS) is an important factor to find changes in images, so that the adjacent pixels are playing a key role in imperceptible of changes in that pixel. The adjacent pixel in a two-dimensional (2-D) image is illustrated in figure4.

Paying attention to Figure4, the image at least must divide to 3×3 blocks, of course the blocks can be selected bigger (4×4 or 5×5), but, considering to the steganography capacity, the size of blocks will choose 3×3.

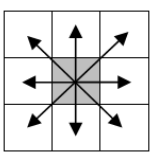


Figure4. Illustration of adjacent pixel in a 2-D image.

In each block only can change the middle-pixel for embedding one bit. Furthermore, in whole blocks could not reverse the middle-pixel, such as blocks with whole white or black color, because the changes in image would be noticeable by an observer.

Resources [11, 12] divide the image to 3×3 blocks like figure (5-a). So that a B&W image in SMS with size (72×28 pixels) is divide to (72×28)/9=216 blocks. Therefore 216 bits of data can be hiding in the image; of course, this quantity is very smaller practically. So that the steganography capacity is very low, for increasing the steganography capacity suggest a new 3×3 blocking method.

Paying attention to the adjacent of each row or column changeable pixels, there are two unchangeable rows or columns, and the only middle-pixel change in each block. So, the two fixed rows or columns can be reduced to one unchangeable row or column by shifting them, figure (5-a). In other words, each block as regard to its upper block, will shift one row to upward and as regard to its left-side block will shift one column to the left. So that data bits can be hidden in image in the form of every other, figure (5-b).

Consequently, the steganography capacity will increase noticeably at most [(72/2) × (28/2)] = 504 bits. Paying attention to the whole blocks is not suitable for embedding data bits, for instance, the middle-pixel flipping in figure 6-b is more tangible than figure 6-a. So, the proper blocks must be selected for storing bits.

Wue et.al [12, 13] proposed a method for selecting flappable blocks in B&W images. In Wue's method, a score is given to each block 3×3. The score which is based on the intangibility of the block leads to the change of the middle-pixel. The scores are between zero to one.

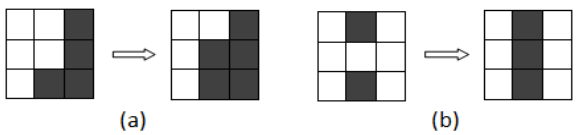


Figure6. The middle-pixel flipping in (a) is less tangible than in (b).

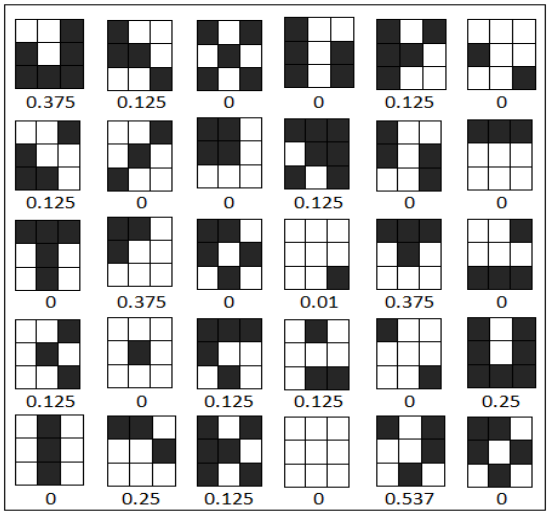


Figure7. A table of some patterns of 3×3 blocks. The bigger scores are indicator of less tangibility to flip middle-pixel.

Zero is indicator of a non-flippable block; and the score which is close to one, indicates that the block is less tangible with flipping middle-pixel. Two features of smoothness and connectivity of pixels are used to obtain the score of each block. The smoothness is estimated by color change in vertical, horizontal and diagonal pixels in the block; and the connectivity is estimated by the B&W clusters number in the block [12, 13]. The number of all obtained conditions for 3×3 blocks with two colors (B&W) equals to $2^9 = 512$. Some of these patterns as well as their scores are illustrated in figure7.

There is a problem in flip-ability algorithm. In the most flippable patterns, with flipping the mid-pixel, the block changes to a non-flippable pattern, figure (8), so that at the time of data extracting, the number of blocks in which the data bits are hidden, will be ignored. Therefore, one cannot access to main information. Furthermore, the numbers of flippable patterns are very few. A method the change in the score of some patterns is that force this problem. For example, with new scoring the right-side blocks are changed to flippable patterns in figures (9-a) and (9-b); and with zero scoring, the left side block is changed to non-flappable pattern in figures (9-c).

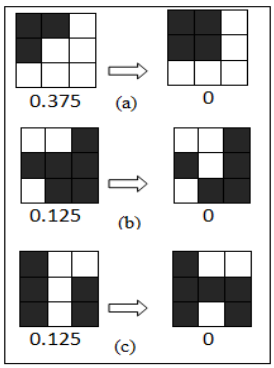


Figure8. By flipping middle-pixel in left-side patterns, in right-side the blocks change to non-flippable patterns.

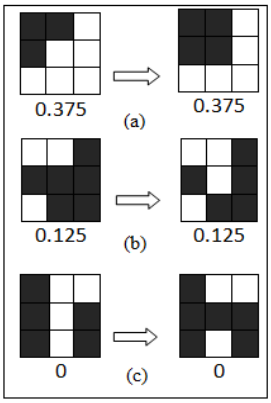


Figure9. Possible changes in flipability score of patterns for efficient extraction.

A method has been proposed by Gyankamel [31] to solve this problem. In this method, flip-ability score of some patterns which is faced to this problem will be quantified by manual scoring. But the numbers of suggested patterns are repeated and very few (about 17 patterns), so that a few data bits can be hidden in image, besides of comparing with pervious methods, the steganography capacity is increased slightly. For increasing capacity, the more patterns should be chosen. It is proposed in this article that with changing score, 18 non-flippable general patterns and their complemented, mirrored and rotated patterns (about 192 patterns) change to flippable patterns and the score of the rest patterns which face to this problem, must change to zero. Some of changed patterns are shown in the figure 10.

Practically, by using of this method the steganography capacity is increased many times of pervious methods. This subject is shown in part 4.

3.2. MAX-MIN method

The easiest way to embed a bit into a block is changing the middle-pixel to the same bit. In binary images, the black color is the indicator of zero bits and the white color of one bit.

In max-min method, at most a bit of data can be hidden in each 3×3 block. To store a bite in each flippable block, at first the number of white and black pixels (except the middle-pixel) is counted, two possible conditions may occur:

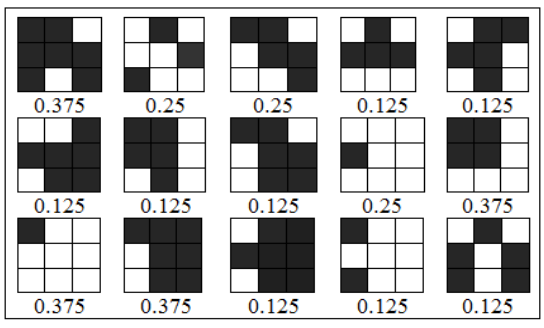


Figure10. The main patterns which is qualified by manual value.

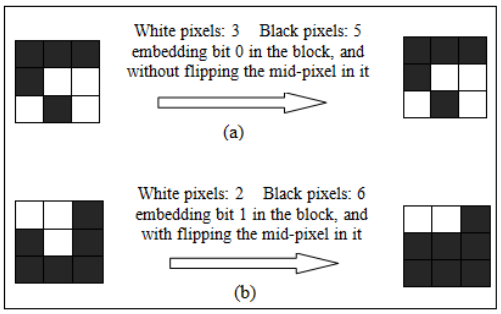


Figure11. Embedding a bit in the block with max-min method.

If the number of white pixels is more than or equal to (>=) black ones, the value of maximum pixel will be allocated to white color and the value of minimum pixel will be allocated to black color.

If the number of White pixels is less than (<) black ones, the value of maximum pixel will be allocated to black color and the value of minimum pixel will be allocated white color.

Now, for hiding a bit with "one value", the color of the middle-pixel must be similar to the color of maximum pixel, i.e. if the maximum pixel in block is black, the middle-pixel color will be black as well, or if the maximum pixel in block is white, the middle-pixel color will be white; therefore, a bit with one value can be hidden in block, figure (11-a).

To hide a bit with "zero value", the color of the middle-pixel must be similar to the color of minimum pixel, i.e.

```
// the determination of the color of maximum and minimum pixels
W-Pixels ← number of white pixels;
B-Pixels ← number of black pixels;

If (W-Pixels > = B-Pixels) {
      maxPixel ← white;
      minPixel ←black;
}
Else {
      maxPixel ← black;
      minPixel ←white;
}

// embedding a bit with value one or zero in the block with using max-min pixels.
If (Bit = = 1)
     Middle-pixel ← maxPixel;
Else
     Middle-pixel ← minPixel;
```

Figure12. The algorithm of max-min method for embedding a data bit in the block

```
// the extraction of a bit of the block by using the color of middle pixel and max-min color.
midPixel ←Color of middle pixel;
If (midPixel = = maxPixel)
     Bit ←1;
Else
     Bit ←0;
```

Figure13. The algorithm of max-min method for extracting a data bit in the block

If the minimum pixel in block is black, the middle-pixel color will be black as well, or if the minimum pixel in block is white, the middle-pixel color will be white; therefore, a bit with zero value can be hidden in block, figure (11-b). The algorithm of this method is shown in figure11.

For data extraction algorithm, the method is opposite of the embedding algorithm. At first, the number of white and black pixels (except the middle-pixel) is counted in the block until the color of maximum and minimum pixels are determined.

Then, if the color of middle pixel is the same as maximum pixel, the hidden bit in the block will be one; otherwise the hidden bit will be zero. This is illustrated in figure 11. This operation is done for all flippable blocks.

The main advantage of this method is that, it is blind steganography method so that some of the bits can store without altering in middle pixel of block, consequently changes would be less and intangible in stego image. Thus, the existence of secret information in the host image is difficult to detect.

3.3. Implemented Algorithm

A brief of implemented approach for data steganography in B&W images in SMS is explained in this article.

Embedding algorithm: First, the B&W image must be changed to an appropriate size (72×28 pixels) and format (ex. WBMP, OTA or OTB) for mobile phones (as it is said in the first of this section). Second, by using of the methods which is said in part 3.1, the image is blocked and the flippable blocks will be determined.

For determination of the flippable blocks, a table of whole of flippable patterns can be created; Although, it is not necessary to store all the models, as simply by one rotate, supplementary and make block contrary in different aspects can be arrival to all the states [10,11].

Now, data bits must embed to the blocks; but before this task, the characters must change to bit values. In pervious methods, each character is converted to eight bits (28=256 characters). But by considering about the possibility of the exchange of the secret message with a few numbers of the characters, there is no need to 256 characters. The method which is suggested here is the using of 64 characters for exchanging of secret message. The

characters are contained all Latin alphabet letters, numbers and even some extra characters or even they can choose arbitrary for more security. So, for embedding each character in image to 6 bits will be needed (64=26), so that by using this method more characters can be hidden in image, therefore the steganography capacity will increase.

After converting the secret message to data bits, the bits will embed to flappable blocks by using MAX-MIN method. Now, the changes store in stego image and then by SMS, the stego image is send to destination.

Surely, for more security before converting to data bits, the secret message can be encrypted by a key – share between sender and receiver – until if the abuser suspicious to exist information to image, he/she could not attain the confidential information.

Extracting algorithm: The data extracting algorithm is the contrary of embedding algorithm. After image blocking and determain of the flappable blocks, with using of data extraction style in MAX-MIN method the data bits are extracted of stego image. Then data bits with use of 6-bit coding method to character until

Now, the information can decode by giving the receiving codes of user. After accessing to secret message, the stego image is deleted for more security.

The main advantage of this approach is that, some of the bits can store without altering any middle pixel block, so that changes would be less and intangible in stego image. This approach is safe from sight attack aspect. So, the security will be a high-level in this steganography method.

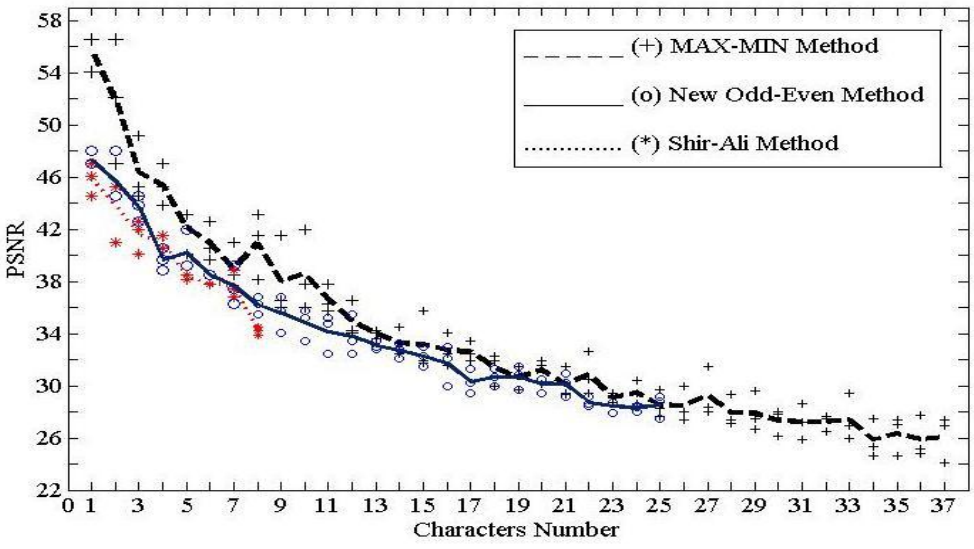
IV. Performance and Evaluation

In order to evaluate proposed method, it is compared to previous methods -New Odd-Even method [3] and Shir-Ali method (stealth steganography) [11]-. The presented method is executed and tested with using the MATLAB software (version 7.0.a). Because, the security and the capacity characteristics are important in steganography, the comparison is based on these features. The security feature is measured with Peak Signal to Noise Ratio (PSNR). PSNR is obtained in equation (4)

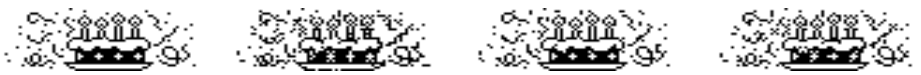
$$PSNR = 10\ell og_{10}\left[\frac{R^2}{MSE}\right] \tag{4}$$

$$MSE = \frac{\sum_{M,N}[I_1(m,n) - I_2(m,n)]^2}{M \times N} \tag{5}$$

In equation (4), Ratio (R) is the given maximum amount in input image pixels. In B&W images, R =1. MSE parameter presents The Mean Square Error between the cover and the stego images, which is calculated in equation (5). In equation (5), I1, I2 are relatively cover image and stego image, and M, N are the dimensions of image pixel. Evaluation is carried out with 20 B&W images of SMS pictures.

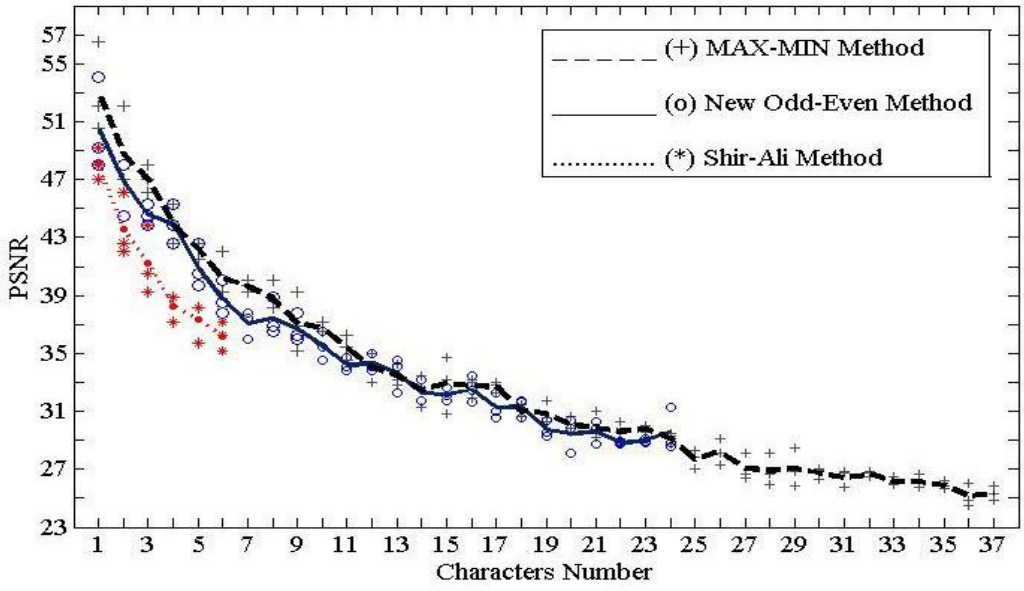


(a)

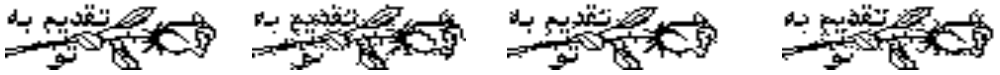


(b) Original Image (c) MAX- MIN Method Image (d) Shir-Ali Method Image (e) New Odd-Even Method Image

Figure14. (a) the comparision MAX-MIN method with Both New odd-even and Shir-Ali methods by PSNR to the number of characters. (b) Cover image, (c) the stego image in Max-Min method with at most 37 hidden characters (d) the stego image in Shir-Ali method with at most 8 hidden characters, (e) the stego image in New Odd-Even method with at most 25 hidden characters.

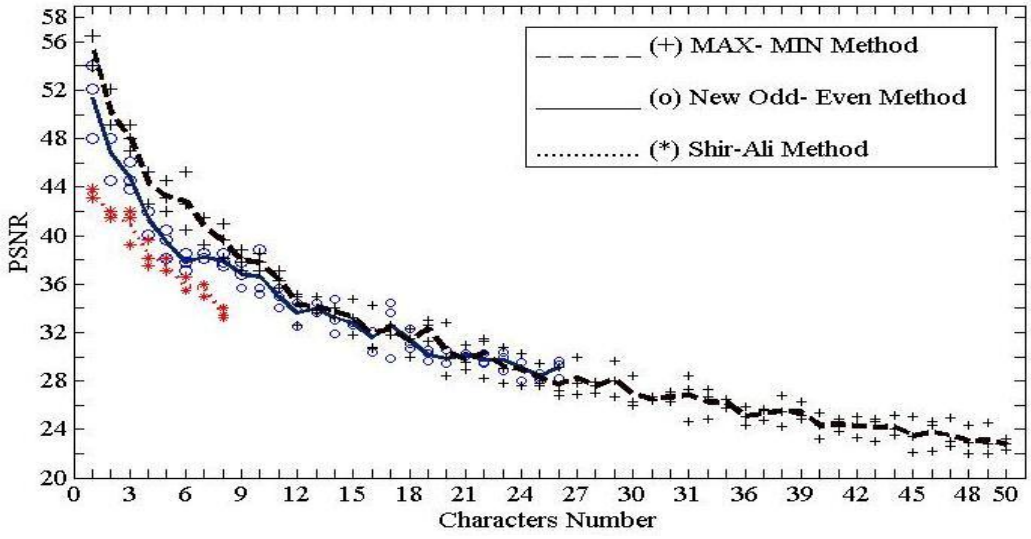


(a)



(b) Original Image (c) MAX- MIN Method Image (d) Shir-Ali Method Image (e) New Odd-Even Method Image

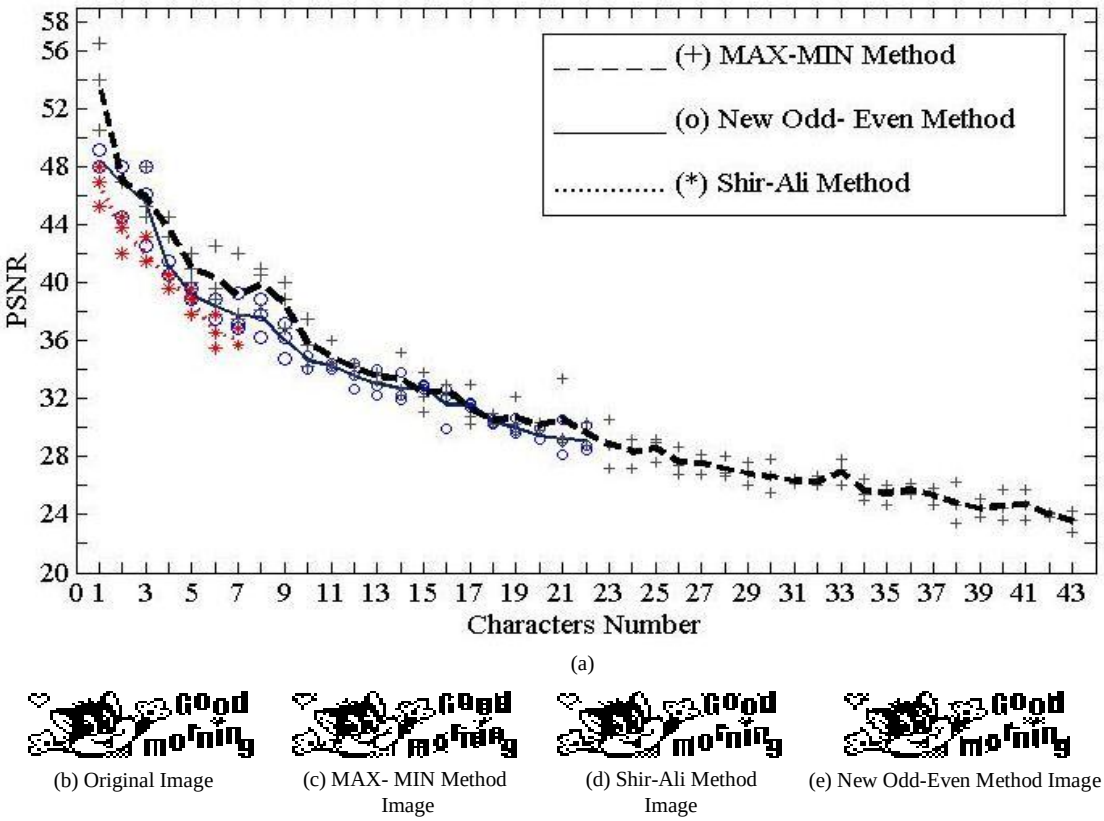
Figure15. (a) the comparison MAX-MIN method with Both New odd-even and Shir-Ali methods by PSNR to the number of characters. (b) Cover image, (c) the stego image in Max-Min method with at most 37 hidden characters (d) the stego image in Shir-Ali method with at most 6 hidden characters, (e) the stego image in New Odd-Even method with at most 24 hidden characters.



(a)

(b) Original Image (c) MAX- MIN Method Image (d) Shir-Ali Method Imag (e) New Odd-Even Method Image

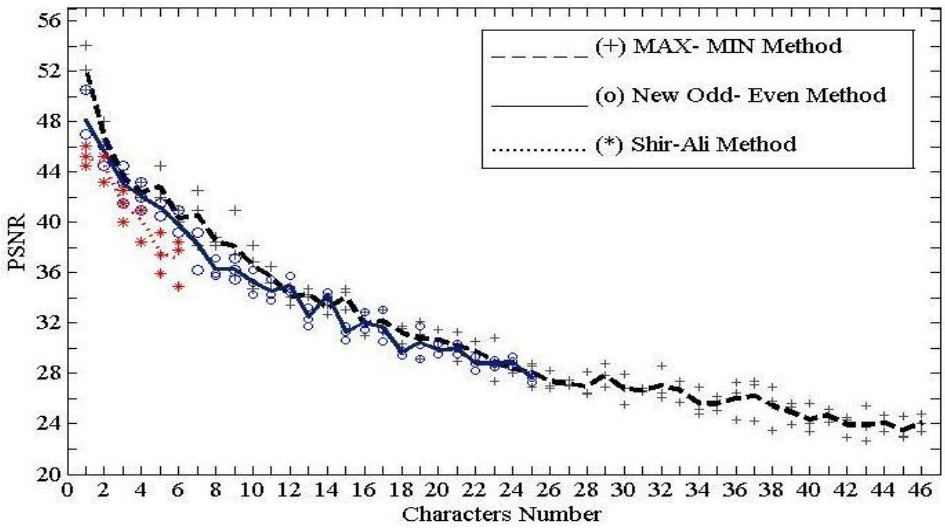
Figure16. (a) the comparision MAX-MIN method with Both New odd-even and Shir-Ali methods by PSNR to the number of characters. (b) Cover image, (c) the stego image in Max-Min method with at most 50 hidden characters (d) the stego image in Shir-Ali method with at most 8 hidden characters, (e) the stego image in New Odd-Even method with at most 26 hidden characters.



(a)

(b) Original Image (c) MAX- MIN Method Image (d) Shir-Ali Method Image (e) New Odd-Even Method Image

Figure17. (a) the comparison MAX-MIN method with Both New odd-even and Shir-Ali methods by PSNR to the number of characters. (b) Cover image, (c) the stego image in Max-Min method with at most 43 hidden characters (d) the stego image in Shir-Ali method with at most 7 hidden characters, (e) the stego image in New Odd-Even method with at most 22 hidden characters.

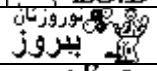
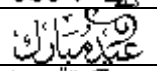
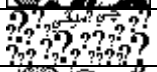
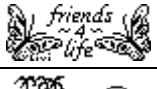
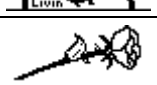


(a)

(b) Original Image (c) MAX- MIN Method Image (d) Shir-Ali Method Image (e) New Odd-Even Method Image

Figure18. (a) the comparison MAX-MIN method with Both New odd-even and Shir-Ali methods by PSNR to the number of characters. (b) Cover image, (c) the stego image in Max-Min method with at most 46 hidden characters (d) the stego image in Shir-Ali method with at most 6 hidden characters, (e) the stego image in New Odd-Even method with at most 25 hidden characters.

Table.4: maximum steganography capacity in 3 methods (Max-Min, New odd-even, Stealth steganography).

Tier	Image	Maximum steganography capacity		
		Shir-Ali method	***New Odd-Even method***	***MAX-MIN method***
1		8	26	46
2		6	21	30
3		8	22	36
4		8	23	35
5		6	24	37
6		9	28	46
7		7	23	44
8		9	25	58
9		6	25	46
10		8	29	53
11		9	29	47
12		6	21	43
13		7	22	43
14		7	25	40
15		4	11	23
16		9	30	53
17		7	25	43
18		8	26	50
19		6	19	41
20		10	32	49
21		5	17	24
22		8	25	37
23		10	29	46
24		8	25	58
The average of steganography capacity in each method		~ 7	~ 24	~ 43

In all experiments, the value of PSNR in MAX-MIN method is more than in both of new odd-even and Shir-Ali methods. For instance, the results of PSNR in three images have been shown in graphs 14-a, 15-a, and 16-a. In graphs (14-18), the maximum amount of hidden capacity has been shown in each method for graphs and images. It is important to know that the more the quantity of PSNR, the less variety in stego image is in relation of cover image. so that the varieties are imperceptible in stego image.

The security feature is measured with the maximum number of hidden character in 20 images which is illustrated in table.4. The average values of maximum capacity in Shir-Ali method is 7 characters. New odd-even method is 24 characters and MAX-MIN method is 43 characters. So, the steganography capacity in MAX-MIN method is higher than pervious methods.

Consequently, by comparing MAX-MIN method with other methods, it offers higher capacity, more efficient imperceptibility, and better security features.

V. STEGANOGRAPHY ATTACKS

The security of the proposed algorithm in binary pictures are considered and it is compared with the other common methods such as POV, OutGuess, F5, Compatibility.

In a F5 attack, hacker cut a 4-pixel bar from the top and left margin of the image, trying to remove the blocked effect of the hiding image and obtain a picture equivalent to the JPEG image, then compress the image and histogram with a suspicious image look into it. Because the Steganography methods are performed on binary images and are not compressed like JPEG images, the image has not been affected due to the lack of a change in the DCT coefficients.

In the Outguess attack, the blocking effect is limited to the 8x8 block boundaries. Due to the fact that in odd and even methods, and the maximum and minimum block is 3 * 3, some blocks cannot be avoided on the 8x8 division boundary. One solution to this is to ignore the rows or pillars of inverted pixels on the 8x8 block boundary. On the other hand, due to the fact that blocks are randomly selected in the image, and because of the use of a few blocks, it is difficult to hide the identity information of the block type.

It should be noted that a large number of message bits are embedded without changing its mid-pixel (without changing the image). This is due to the existence of even and odd or a maximum and minimum in the storage of bits. On the other hand, pixels that have been changed in a number of blocks may vary with the amount of time that this block is embedded.

As an example, with odd and even method which is shown in figure 19, seventeen characters are hidden in the image. In order to embed seventeen characters, one hundred and eight pixels are required for embedding, but in the figure 19 only forty-seven pixels are changed and the remaining sixty-one remain unchanged in pixels according to the Steganography method are saved.

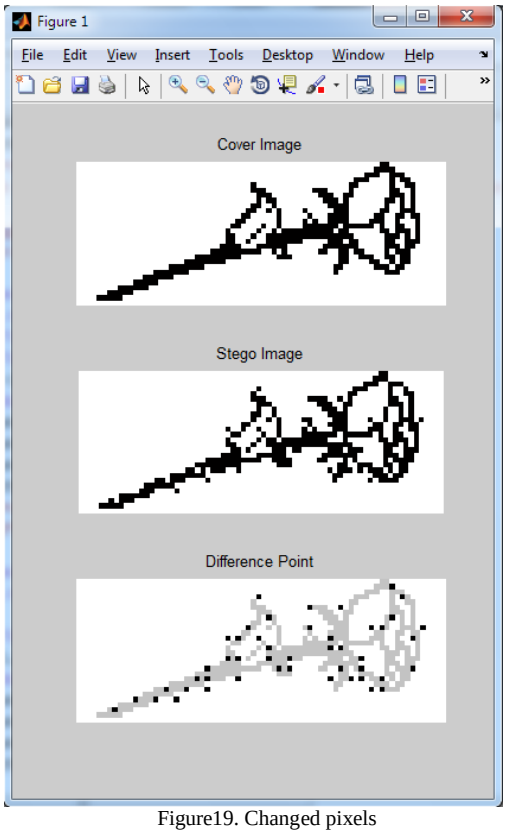


Figure19. Changed pixels

A security key is used between the sender and receiver during cryptographic operation and information extraction, using this key a random sequence of pixels suitable for concealment Binary image is selected. This key causes the cryptographic method to fail to access information if it is attack. In this regard, the security of the proposed methods is increased. JPEG Compatibility attack works only on images that use pixel-level data hiding and compressed in JPEG format. Since the images used in software developed in binary format are stored and used, the attack does not affect these images.

VI. IMPLEMENTATION OF THE PROPOSED METHOD

The proposed method has been implemented and tested using MATLAB software. Then, in order to use this method in the practical cases, it was implemented and developed by the Java programming language, J2ME Programming Platform (J2ME). The program was installed on several mobile phones such as Nokia C3-00, Nokia 6330, Nokia 2700, SonyEricson K800. Figure 20 and figure 21 show the implementation of the proposed algorithm.

The software was produced in two versions for the odd and even methods, and the maximum and minimum methods. Since in the methods of concealment of two algorithms, a message embedding algorithm in an image and an algorithm for extraction of information from the image, each software version consists of two separate algorithms (the Steganography algorithm and extraction algorithm).

Figure20. Implementation of Steganography algorithm on sending device

Figure21. Extracted of the information on receiving device

VII. CONCLUSION

In this article, the Max-Min approach is introduced as a blind steganography method for hiding the textual data in B&W images of SMS. Furthermore, by presentation of a new style in image blocking, selection methodology blocks and alternation in the way of character coding, the steganography capacity is significantly increased. The Max-Min method is related to color of central pixel and the number of black and white pixel in each block. In fact, the only center pixel value may be changed in a block and only a bit can be hidden in it. The hidden secret message is recovered with investigating central pixel with the rest of the pixels in block; and so, there is no need to have the original image. By comparing Max-Min method with other methods, our new method offers higher capacity, more efficient imperceptibility, and better security features.

REFERENCES

[1] Jindal, Sakshi, and Navdeep Kaur. "Digital image steganography survey and analysis of current methods." International Journal of Computer Science and Information Technology & Security 6 (2016).

[2] Ingcmar J. Cox, Mattew L. Millet, Jeffrey A. Bloom, Jessica Fridrich, and Tan Kalker, Digital Watermarking and Steganography,2th Ed. The Morgan kaufman publishers, USA, Elsevier, 2008.

[3] A.zare, A.R.Naghshnilchi "Steganography in SMS by New odd-even method" , Proc. of the 16th International CSI Computer Conference (CSICC'2006), School of Computer Science, IPM, Tehran, Iran, 24-26 Jan. 2011, pp.905-910 , (in Persian).

[4] Shirali-Shahreza, M.H., Shirali-Shahreza, M., "Steganography in SMS by Sudoku Puzzle", 978-1-4244-1968-2008 IEEE.

[5] Shirali-Shahreza, M, "Text Steganography by Changing Words Spelling", In 10th International Conference on Advanced Communication Technology, ICACT 08, vol. 3, pp. 1912-1913, 17-20 Feb. 2008

[6] Shirali-Shahreza, M.H., Shirali-Shahreza, M., "Text Steganography in Chat", In 3rd IEEE/IFIP International Conference in Central Asia, pp. 1-5, 26-28 Sept. 2007

[7] M.H. Shirali-Shahreza and M. Shirali-Shahreza, "A New Approach to Persian/Arabic Text Steganography", Proceedings of the 5th IEEE/ACIS International Conference on Computer and Information Science (ICIS 2006), Honolulu, HI, USA, 10-12 July, 2006, pp. 310-315.

[8] M. Shirali-Shahreza, "Steganography in SMS," Proc. of the 11th International CSI Computer Conference (CSICC'2006), School of Computer Science, IPM, Tehran, Iran, 24-26 Jan. 2006, pp.905-910 , (in Persian).

[9] JohnWiley and Sons Ltd, Mobile Messaging Technologies and services SMS,EMS and MMS,2nd Ed. Gwenael Le Roodic,England.,West Sussex Po19 8SQ,2005.

[10] N.F. Johnson, S.C. Katzenbeisser, A survey of steganographic techniques, in: S. Katzenbeisser, F.A.P. Petitcolas (Eds.), Information Hiding Techniques for Steganography and Digital Watermarking, Artech House, Inc., Norwood, 2000.

[11] M. Shirali-Shahreza, "Stealth Steganography in SMS,"Proceedings of the third IEEE and IFIP International Conference on Wireless and Optical Communications Networks (WOCN 2006), 11-13 April, 2006.

[12] M. Wu, E. Tang, and B. Liu, "Data hiding in digital binary image," in IEEE Int. Conf. Multimedia & Expo, New York, 2000.

[13] M. Wu and B. Liu, "Data Hiding in Binary Image for Authentication and Annotation," IEEE Trans. on Multimedia, vol. 6, no. 4, August 2004, pp.528-538.

[14] M. Shirali-Shahreza, "An Improved Method for Steganography on Mobile Phone," WSEAS Transactions on Systems, Vol. 4, Issue 7, July 2005, pp. 955-957.

[15] Soria-Lorente, A., and S. Berres. "A Secure Steganographic Algorithm Based on Frequency Domain for the Transmission of Hidden Information." Security and Communication Networks 2017 (2017).

[16] Provos, N., "Defending Against Statistical Steganalysis", Proc. 10th Usenix Security Symp, Usenix Assoc., pp. 323-335, 2001.

[17] Lee, Y.K. Chen, L.H. "High capacity image steganographic model," Vision, Image and Signal Processing, IEE Proceedings, Vol 147, 288-294, Jun 2000.

[18] Anderson, R.J, Petitcolas, F.A.P., "On the Limits of Steganography", IEEE Journal of Selected Areas in Communications, Special Issue on Copyright and privacy Protection, Vol. 16(4), pp. 474-481, May 1998.

[19] Y. C. Tseng, Y. Y. Chen, and H. K. Pan, "A Secure Data Hiding Scheme for Binary Images," IEEE Trans. on Communications, Vol. 50, No. 8, Aug. 2002, pp. 1227-3 1.

[20] Y. Y. Chen, H. K. Pan, and Y. C. Tseng, "A Secure Data Hiding Scheme for Two-Color Images," IEEE Symposium on Computers and Communications, 2000, pp. 750-755.

[21] Y. C. Tseng and H. K. Pan, "Secure and Invisible Data Hiding in 2-Color Images," IEEE INFOCOM, 2001, pp. 887-896.

[22] Rafat K.F., "Enhanced Text Steganography in SMS", 2nd International Conference on Computer, Control and Communication, 18 Feb 2009.

[23] Cole E.," Hiding in Plain Sight: Steganography and the Art of Covert Communication", John Wiley, Indianapolis, Indiana, 2003.

[24] Venkatesan M., MeenakshiDevi P., Duraiswamy K., Thiagarajah K., "A New Data Hiding Scheme with Quality Control for Binary Images Using Block Parity" Third International Symposium on Information Assurance and Security, IEEE, 2007.

[25] Chhajed J., Inamdar V., Attar V., "Steganography in Black and White Picture Images", Congress on Image and Signal Processing, IEEE, 2008.

[26] Böhme R., "Advanced Statistical Steganalysis", Springer, Verlag Berlin Heidelberg, 2010.

[27] Bender W., Gruhl D., Morimoto N., Lu A., "Techniques for data hiding", IBM Systems Journal, vol. 35, Issues 3&4, 1996, pp. 313-336.

[28] Kim Y., Moon K., Oh I., "A Text Watermarking Algorithm based on Word Classification and Inter-word Space Statistics", Proceedings of the Seventh International Conference on Document Analysis and Recognition (ICDAR'03), 2003, pp. 775–779.

[29] Gutub A., Fattani M., "A Novel Arabic Text Steganography Method Using Letter Points and Extensions", Proceedings of the WASET International Conference on Computer, Information and Systems Science and Engineering (ICCISSE), Vienna, Austria, Vol. 21, 2007, pp. 28-31.

[30] Aabed M.A., Awaideh S.M., Elshafei A.M., GutuA. A. b, "Arabic Diacritics Based Steganography", Proceedings of the International Conference on Signal Processing and Communications (ICSPC 2007), Dubai, UAE, 2007, pp. 756-759.

[31] Gyankamal J. Chhajed ; Vandana Inamdar ; Vahida Attar, " Steganography in Black and White Picture Images" IEEE Image and Signal Processing, CISP '08. Congress on Sanya, Hainan, China, 27-30 May 2008.

[32] Ajetrao H., Kulkarni P.J., Gaikwad N., "A Novel Scheme of Data Hiding in Binary Images", International Conference on Computational Intelligence and Multimedia Applications, IEEE, 2007.

[33] Prem Singh, Rajat Chaudhary and Ambika Agarwal " A Novel Approach of text Steganography based on null spaces" IOSr Journal of Computer Engineering (IOSRJCE) ISSN:2278-0661 Vol 3, Issue 4 (July – Aug 2012) PP 11-17 www.isorjournals.org

[34] Thamaraiselvan, R., and A. Saradha. "A Novel approach of Hybrid Method of Hiding the Text Information Using Stegnography." IJCER 1, no. 1 (2012).

[35] Zhi-Hui Wang " Emoticon-Based Text Steganography in Chat" School Of Software Dalian University of Technology Dalian,Lionging China, 978-1-4244-4607-0/09/25.00 ©2009 IEEE wangzhihui

[36] Ahmed khan, "Robust Textual Steganography", Journal of Science (JOS) ISSN 2324-9854, Vol. 4, No. 4, 2015, Pages: 426-434.

[37] Shunquan Tan , Haojie Zhang , Bin Li , Jiwu Huang. "Pixel-Decimation-Assisted Steganalysis of Synchronize-Embedding-Changes Steganography", IEEE Transactions on Information Forensics and Security (Volume: 12, Issue: 7, July 2017). Pages: 1658 – 1670.