Context Sensitive Steganography on Hexagonal Interactive System

T. Nancy Dora, S. M. Saroja T. Kalavathy(✉), and P. Helen Chandra

Jayaraj Annapackiam College for Women (Autonomous), Theni District,
Periyakulam, Tamilnadu, India
nancydora.t@gmail.com, kalaoliver@gmail.com, chanrajac@yahoo.com

Abstract. Cryptography is the science of using mathematics to encrypt and decrypt data and Steganography is the art and science of hiding communication. A steganographic system thus embeds hidden content in unremarkable cover media so as not to arouse an eavesdropper's suspicion. The Steganography hides the message so it cannot be seen. In this paper, a new method is proposed to embed data in images. The security is provided through context sensitive rules. Hexagonal Finite Interactive System is taken as a base to choose the Carrier. Experimental results show that the method is very efficient especially when gluing is done by matching the border label so that the tiling is done uniformly.

Keywords: Image steganography · Secret image
Context sensitive stego technique · Stego image · Secret key

1 Introduction

Data security is a challenging issue of data communications today that touches many areas including secure communication channel, strong data encryption technique and trusted third party to maintain the database. The rapid development in information technology, the secure transmission of confidential data herewith gets a great deal of attention. The conventional methods of encryption can only maintain the data security. The information could be accessed by the unauthorized user for malicious purpose. Therefore, it is necessary to apply effective encryption/decryption methods to enhance data security. The two important techniques for providing security are cryptography and steganography [Mi1]. Both are well known and widely used methods in information security.

Steganography has been progressively becoming one of the popular technique to be used for secret communication between two parties or more. The term of steganography originated from two Greek words which were stegano and graphos. Stegano could be described as cover or secret and graphos defined the meaning of writing or drawing. The combination of both words delineated the meaning of "covered writing" [SK1].

© Springer Nature Singapore Pte Ltd. 2018
D. Ghosh et al. (Eds.): ICMC 2018, CCIS 834, pp. 16–26, 2018.
https://doi.org/10.1007/978-981-13-0023-3_2

Several steganography methods began to propose embedded secret message in multimedia objects such as images. Images could be a powerful host to hide information because of the spacious spaces it offers. Moreover, the changes in digital images are usually unnoticeable to naked eye. Nowadays, computer technology has given a new life to the ancient steganography. Computer technology introduces digital steganography and makes the steganography easier to execute but harder to crack. These facts stimulate to propose a new model for hiding text in image. A tiling system generates a grid of tiles using the set R of production. Another type of steganographic scheme based on context-sensitive tilings is explored in [PR1] rules.

A novel method of hiding text in image is defined and implemented through tiling. A context sensitive rule is used to generate the Stego image. The images thus produced can be said to have both structure and semantics. A new algorithm has been proposed that would satisfy all the principles of security i.e. confidentiality, authentication, integrity and non-repudiation and also satisfy the requirements of steganography i.e. capacity, undetectability and robustness.

2 Preliminaries

2.1 Steganography [KK1,MB1]

Steganography is the technique of embedding hidden messages in such a way that no one, except the sender and intended receiver(s) can detect the existence of the messages. The main goal of steganography is to hide the secret message or information in such a way that eavesdroppers are not able to detect it [NS1]. Other goal of steganography is to communicate securely in a completely undetectable manner. The various forms of data in steganography can be audio, video, text and images etc.

The basic model of Steganography consists of three components: The Carrier image: The carrier image is also called the cover object that will carry the message that is to be hidden. The Message: A message can be anything like data, file or image etc. The Key: A key is used to decode/decipher/discover the hidden message.

Steganography can be achieved when the user can retrieve a secret message unnoticeably. This involves two main processes. The first process is embedding process, where a secret message is embedded in the host. The host and a secret message can be an image, a video, an audio or text. The second process is involving the extraction of the secret message that has been embedded. Generally steganography concepts can be represented by a basic model of steganography as in Fig. 1.

Various types of Steganography include Image Steganography, Audio Steganography, Video Steganography, Text files Steganography, etc. The image steganography is the process in which we hide the data within an image so that there will not be any perceivable change in the original image. Different techniques of Steganography like Least Significant Bit and Bitmap Steganography are available. Recently, image has been used in steganography as a carrier to

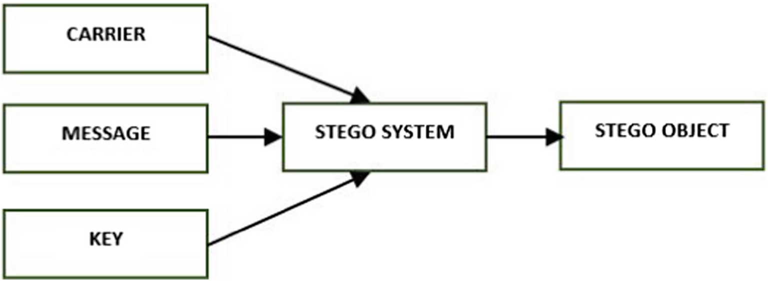

Fig. 1. Basic model of steganography

transmit or send the secret message from a sender to a receiver. The reason is because a huge amount of information can be hidden without noticeable impact to the image that is used as carrier. In addition, the usage of image in information hiding is an ideal technique to have a secured steganography because digital image is insensitive to human visual system.

We are also looking back the definition of hexagonal tiles and scenarios designed by tiling hexagonal unit cells with colors representing two dimensional hexagonal pictures and the structure of a complete 3 × 3 Hexagonal Finite Interactive System ($HFIS$) [NA1].

2.2 Hexagonal Grid

Hexagonal grid is an alternative representation of pixel tessellation scheme for the conventional square grid for sampling and representing discredited images. Each pixel is represented by a horizontal deflection followed by a deflection upward and to the right. These directions are represented by a pair of unit vectors u and v and this coordinate system is referred as the "h_2" system. Given a pixel with coordinates (u, v) (assumed integer), the coordinates of the neighbors are illustrated in Fig. 2.

2.3 Hexagonal Tiles and Scenarios

Let Σ be a finite alphabet. A hexagonal tile is a hexagonal cell labelled with symbol from the given alphabet and enriched with additional information on each border. This information is represented abstractly as an element from a finite set and is called a border label. The role of border labels is to impose local gluing constraints on self-assembling tiles: two neighbouring cells, sharing a side border (east-west or north east-south west or north west-south east) should agree on the label on that border. A hexagonal scenario is similar to a two-dimensional hexagonal picture, but: (1) each hexagonal cell is replaced by a tile; and (2) east-west or north east-south west or north west-south east neighbouring cells have the same label on the common border.

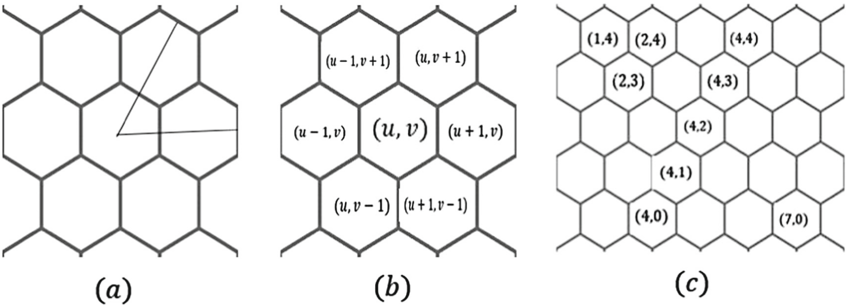

Fig. 2. (a) A coordinate system based on unit vectors u and v, (b) the neighborhood of a hexagonal pixel and (c) labeling of a hexagonal pixel

Graphically, a hexagonal scenario is obtained using the hexagonal tiles representing the transitions and identifying the matching classes or states of the neighbouring cells. The labels on the north east and north west borders represent north memory states, while the south east and south west borders represent south memory states and the ones on the west and east borders represent interaction classes. The selected labels on the external borders are called initial for south west, west and north west borders and final for north east, east and south east borders.

To construct the hexagonal scenarios by assembling hexagonal tiles, we make use of the following. N is the set of natural numbers $\{0, 1, 2, ...\}$, $Z = N \cup -N$ is the set of integers and R is the set of real numbers. We will be working in the two-dimensional hexagonal grid of integer positions $Z \times Z$. The directions $\mathcal{D} = \{EE, WW, NE, NW, SE, SW\}$ will be used as functions from $Z \times Z$ to $Z \times Z$: A point on the side borders in a unit cell is specified by its middle points such that $EE(x, y) = (x + 1, y), WW(x, y) = (x - 1, y), NE(x, y) = (x, y + 1), NW(x, y) = (x-1, y+1), SE(x, y) = (x+1, y-1)$ and $SW(x, y) = (x, y-1)$. We say that (x, y) and $(x^{'}, y^{'})$ are neighbors if $(x^{'}, y^{'}) \in \{EE(x, y)/WW(x, y)/NE(x, y)/NW(x, y)/SE(x, y)/SW(x, y)\}$. Note that $EE = WW^{-1}, NE = SW^{-1}$ and $NW = SE^{-1}$. Examples of tiles and scenarios are presented in Fig. 3.

2.4 Hexagonal Finite Interactive System (HFIS)

Let Σ be a finite alphabet. A Hexagonal Finite Interactive System **(HFIS)** over Σ is defined by: a set $S = s_1, s_2, s_3, s_4$ of states and a set $C = c_1, c_2$ of classes; a set T of transitions of the form: $(s_1, c_1, s_2) \rightarrow a \rightarrow (s_3, c_2, s_4)$ where a is a symbol of a given alphabet Σ; specification of the initial/final states and classes. Let s_a be the scenario of direct transition where a is the labelled symbol over the alphabet. The set of all scenarios representing hexagonal picture is denoted by $\Sigma^{**}_{s_2}$.

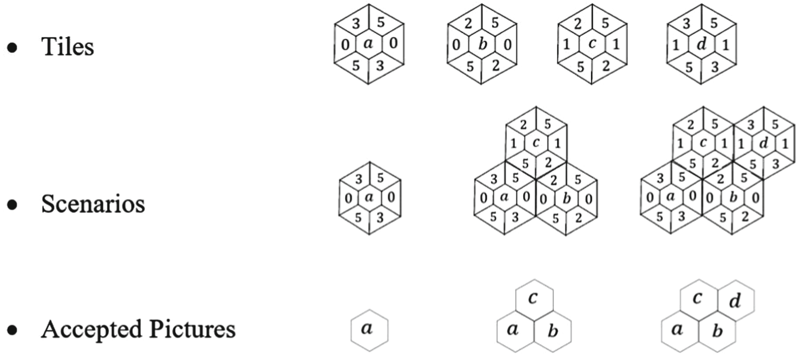

Fig. 3. Tiles and scenarios

A *HFIS* is complete if it specifies a transition $(s_1, c_1, s_2) \rightarrow t \rightarrow (s_3, c_2, s_4)$ for any pair $((s_1, c_1, s_2), (s_3, c_2, s_4))$ in $((S \times C \times S) \times (S \times C \times S))$. A tile representation is used which is based on showing the transitions and stating which states and classes are initial/final. The states/classes of this *HFIS* is denoted by the initials of the colors: The classes c_1 and c_2 are g (*green*) and b (*blue*) while the swne memory states s_1 and s_3 are p (*purple*) and m (*magenta*) and the nwse memory states s_2 and s_4 are r (*red*) and o (*orange*). A scenario is called indecomposable if all its south west and north east borders are labelled with s_1 and s_3 west and east borders with c_1 and c_2 and the north west and south east borders with s_2 and s_4 respectively and it doesn't contain any sub-scenarios with this property. A complete 3×3 *HFIS* is specified by the 64 transitions shown in Fig. 4.

3 Steganography Through *HFIS*

A New Model for Hiding Text in an image using Image Steganography through Hexagonal Finite Interactive System is proposed. An interactive system *HFIS* recognizing the Hexagonal grid consisting of a parallelogram array with empty tokens is considered as a carrier to carry the image. A simple and an efficient model based on context sensitive classes/states (CSC/CSS) replacement technique is stretched out for calculating secret message that can be embedded in an image. The embedding process distribute the secret message inside a shared colored images.

3.1 Context Sensitive Classes/States (*CSC*/*CSS*) Replacement Technique

In image steganography techniques, this proposed model uses substitution technique. *CSC* replaces the color of the class in the image cell and *CSS* replaces the color of the state in the image cell.

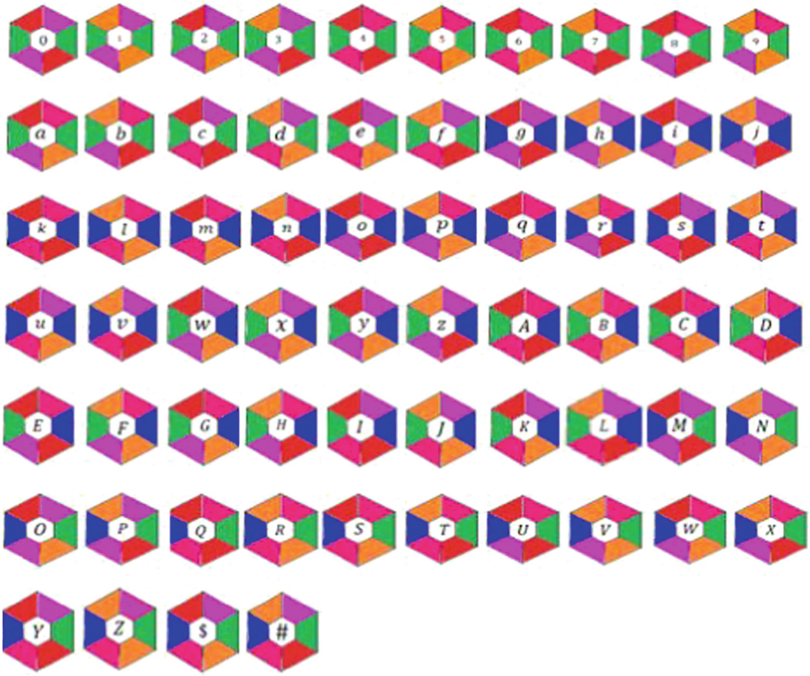

Fig. 4. Transitions of 3 × 3 *HFIS* (Color figure online)

Each character of the original text is represented by a tile with border rule representing the action/transition of the unit cell. Each tile is a unit cell surrounded by two classes and four states The transition is directed by a context sensitive rule $(x_1\alpha_1\gamma_1 y_1\beta_1\delta_1) \Rightarrow (x_2\alpha_2\gamma_2 y_2\beta_2\delta_2)$. The graphical representation of the rule is shown in Fig. 5.

$\{x_i, y_i\}$ represents the classes either b $(blue)$ or g $(green)$; $\{\alpha_i, \beta_i\}$ represents the swne memory states either m $(magenda)$ or p $(pink)$; $\{\gamma_i, \delta_i\}$ represents the nwse memory states either o $(orange)$ or r (red).

The *HFIS* depends on the constraints of the transitions identifying the matching classes or states of the neighboring cells. The three sensitive class/states of each character in the secret message is represented by three bits of zeros. If there is a mismatching of class or states in a character, the particular character is sensitive. Usually the class representing x_1 and the states representing α_1 and δ_1 are more sensitive than the others. These sensitive class and states are modified to hide the character of that cell. In that case, the corresponding color is modified so that the original character is concealed in the cell and the identified bit is changed from zero to one. The collective bits are converted into decimal value to find the secret key. The conversion is done by choice of taking bits of length less than twenty one at a time. We may develop a secret key for each row. The assembled scenario is considered as the stego image in which the

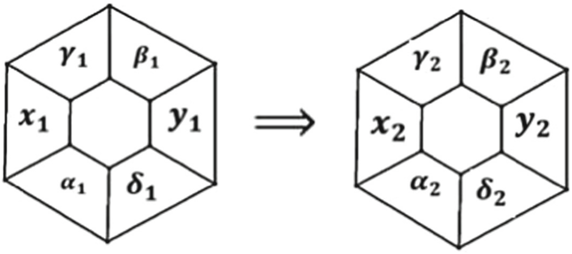

Fig. 5. Transition rule

secret message is concealed. The uniqueness of the data distribution process, made this technique resistant to the attacks as it is difficult for the attackers to reconstruct the shape from stego images. The carrier, the stego image and the secret key together are called as context sensitive stego object and the process of getting stego object is called as context sensitive stego system.

3.2 Steganography Algorithm for Encryption: (Secret Image to Stego Image)

Input: $SecretText, SecretImage, Carrier$
Output: $Secretkey, StegoImage$
Algorithm:

```
1. Fix the number of rows as $I$ and identify the row as $Row(I)$
2. Calculate the length of the S$ecretText$ in each row and fix it as $textlength(Row(I))$
3. Let $StegoImage = EmptyGrid(Carrier)$
4. For each $Row(I)$ do the following
5. Place the SecretImage of each characters one by one in the Grid as follows
6. Fix the sensitive class and states of the Current Image
7. Check the sensitive class/ states with the StegoImage
8. If $maching = ok$ assemble with StegoImage and goto step 5
9. If not, modify the unmatched class and states and assemble the image with StegoImage
10. Modify $x\alpha\delta$ and goto step 5
11. Convert $x\alpha\delta$ of $row(I)$ into decimal ; fix it as $Secretkey(I)$ and goto step 4
12. Get final StegoImage and Secret Key as output
```

3.3 Implementation of Context Sensitive Steganography

The following example elucidates the algorithm. Secret Image of the 64 characters are stored in a stack. The authors wish to send 'GOOD, VERY GOOD, and EXCELLENT' as a secret message and choose the Hexagonal grid consisting of a

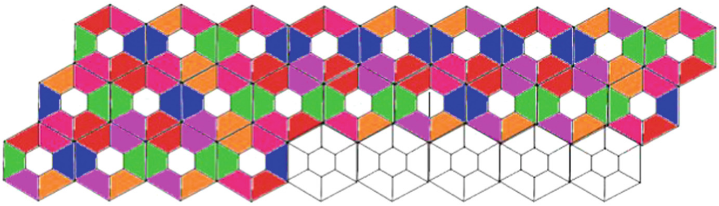

Fig. 6. Stego image

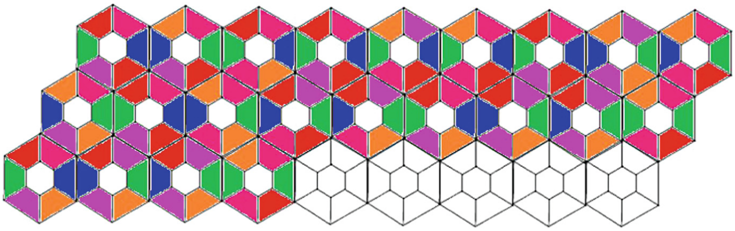

Fig. 7. Secret image

parallelogram array with empty tokens on a hexagonal finite interactive system as a Carrier to carry the image as a cover object. "GOOD" will be assembled by four transition tiles in the first row, "VERY GOOD" will be assembled by nine transition tiles in the second row and "EXCELLENT" will be assembled by nine transition tiles in the third row. The minimum required size of the secret image is 3 × 9. Choose a cover object of size 3 × 9.

Now the characters in the Secret Text is given as input and the Context Sensitive Classes/States (CSC/CSS) Replacement Technique is applied. The transitions for "GOOD", "VERY GOOD" and "EXCELLENT" are modified. The process of modification is given in table 1 (Fig. 8). The modified tiles are assembled in the cover object as a hexagonal scenario. The scenario is considered as the Stego Image. The identified bits of the sensitive classes and states are modified and collected row wise. The modified bits for the first, second and third row are respectively 000 000 100 000, 011 000 010 100 100 000 000 100 000 and 010 011 001 100 110 100 100 000 100. They are converted into decimal value and the corresponding Secret keys are obtained. The Secret keys are 32, 51003424, 40265988. The final Stego Image is shown in Fig. 6.

Now, Stego image and secret keys are sent to the receiver. The receiver recognize the character of the Stego Image row wise. The Secret keys are converted into binary digits. The receiver now identify the modified bits and the corresponding modified state or class of each tile. The identified class or states are remodified and the corresponding Secret image is received as shown in Fig. 7. The equivalent secret text is obtained.

secret Character	Secret Transition	secret Image	CSS/CSC Rule	Modified Transition	Stego Image	Modified Bits($x\alpha\delta$)	Stego Character
First row							
G	*pgrmbo*		-	-		000	G
O	*pbrpgo*		-	-		000	O
O	*pbrpgo*		$b \rightarrow g$	*pgrpgo*		100	2
D	*mgombr*		-	-		000	D
Second row							
V	*pbomgo*		$p \rightarrow m$ $o \rightarrow r$	*mbomgr*		011	T
E	*pgrmbr*		-	-		000	E
R	*mbomgo*		$m \rightarrow p$	*pbomgo*		010	V
Y	*mbrpgr*		$b \rightarrow g$	*mgrpgr*		100	c
space	*mbrpgo*		$b \rightarrow g$	*mgrpgo*		100	e
G	*pgrmbo*		-	-		000	G
O	*pbrpgo*		-	-		000	O
O	*pbrpgo*		$b \rightarrow g$	*pgrpgo*		100	2
D	*mgombr*		-	-		000	D
Third row							
E	*pgrmbr*		$p \rightarrow m$	*mgrmbr*		010	A
X	*pbomgr*		$p \rightarrow m$ $r \rightarrow o$	*mbomgo*		011	R
C	*mgrmbo*		$o \rightarrow r$	*mgrmbr*		001	A
E	*pgrmbr*		$g \rightarrow b$	*pbrmbr*		100	o
L	*mgopbr*		$g \rightarrow b$ $m \rightarrow p$	*pbopbr*		110	j
L	*mgopbr*		$g \rightarrow b$	*mbopbr*		100	v
E	*pgrmbr*		$g \rightarrow b$	*pbrmbr*		100	o
N	*pbopgo*		-	*mbopgo*		000	N
T	*mbomgr*		$b \rightarrow g$	*mgomgr*		100	7

Fig. 8. Process of secret image to stego image

3.4 Steganography Algorithm for Decryption: (Stego Image to Secret Image)

Input: $Secretkey, StegoImage, Carrier$
Output: $Secretkey, SecretImage$
Algorithm:

1. Fix the number of rows as I and identify the row as $Row(I)$
2. Calculate the length of the $StegoText$ in each row and fix it as $textlength(Row(I))$
3. Let $SecretImage = EmptyGrid(Carrier)$
4. For each $Row(I)$ do the following
5. convert the Secret key of Row(I) to binary digit
6. Extract the binary digit in bits of length 3 and store it in an array BIN
7. For each StegoImage of Row(I) do the following
8. Let $J = J^{th}$ tile in $Row(I)$
9. If $BIN(J) = 000$, place the Stego Image in the carrier.
10. If not, Identify the modified bits and corresponding modified states or class
11. Modify the stego transition into secret transition and place it in the carrier and goto step 7.
12. Goto Step 4 and do the process of next Row:
13. Get final SecretImage as output from the Carrier.

4 Conclusion

We have presented a stego-system which generates stego-objects using context sensitive tiling. A new steganographic algorithm for hiding text in images is proposed. This new steganographic approach is robust and very efficient for hiding text in images. We have further planned to develop the system in java based on the proposed algorithm. Steganography will continue to increase in popularity over cryptography. The system would be tested on the basis of various illustrations and the results would be compared with those of existing algorithms.

References

[KK1] Rahmani, M.K.I., Arora, K., Pal, N.: A crypto-steganography: a survey. Int. J. Adv. Comput. Sci. Appl. **5**(7), 149–155 (2014)

[MB1] Maiti, C., Baksi, D., Zamider, I., Gorai, P., Kisku, D.R.: Data hiding in images using some efficient steganography techniques. In: Kim, T., Adeli, H., Ramos, C., Kang, B.-H. (eds.) SIP 2011. CCIS, vol. 260, pp. 195–203. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-27183-0_21

[Mi1] Rajyaguru, M.H.: Crystography - combination of cryptography and steganography with rapidly changing keys. Int. J. Emerg. Technol. Adv. Eng. **2**, 329–332 (2012)

[NA1] Nancy Dora, T., Athisaya Ponmani, S., Helen Chandra, P., Kalavathy, S.M.S.T.: Generation of hexagonal patterns in finite interactive system and scenarios. Glob. J. Pure Appl. Math. **13**(5), 17–26 (2017)

[NS1] Johnson, N.F., Jajodia, S.: Exploring steganography: seeing the unseen. Computer **31**, 26–34 (1998)

[PR1] Ritchey, P.C., Rego, V.J.: A context sensitive tiling system for information hiding. J. Inf. Hiding Multimedia Sig. Process. **3**(3), 212–226 (2012)

[SK1] Sharma, V., Kumar, S.: A new approach to hide text in images using steganography. Int. J. Adv. Res. Comput. Sci. Soft. Eng. **3**(4), 701–708 (2013)