**Image Steganography using Sudoku: A Combined Approach**

Shreekari

shreekari.16cs081@sode-edu.in

Shreya

shreya.16cs084@sode-edu.in

Shreya Bhat

shreya.16cs085@sode-edu.in

Rama Moorthy H

ramamoorthy.cs@sode-edu.in

***Department of Computer Science and Engineering,***

***Shri Madhwa Vadiraja Institute of Technology Management, Bantakal****,* ***India.***

***Abstract:*** *Steganography is the art of hiding information within non-secret, ordinary file or message in order to avoid detection of the secret to be shared. The original work was proposed by Chang et al., it was later modified by Hong et al. and Roshan Shetty et al. The proposed method combines the method given by Hong et al. and Roshan Shetty et al., here a better position for data embedding can be selected to minimize image distortions also the embedding capacity is increased by using color image. The half-filled Sudoku puzzle is shared between sender and receiver. Sender will use this puzzle to embed the secret message in the image and the receiver will use same puzzle to extract the secret message.*

***Keywords:*** *Sudoku puzzle, Image Steganography, Data hiding, Reference Matrix, RGB Image, pixels, Secret data.*

1. **INTRODUCTION**

The purpose of data hiding is to enhance communication security by embedding secret messages into digital media, especially digital images so that the existence of the messages is not apparent. Using digital images as cover media to conceal secret data is an important issue for secret data delivery applications. The existing schemes of data hiding can roughly be classified into the following 3 categories [2]

Spatial domain data hiding: Data hiding of this type directly modifies image pixels in the spatial domain for data embedding. This technique is easy to implement, offers a relatively high hiding capacity, and the quality of the stego image can be easily controlled. Therefore, data hiding of this type has become a popular method for image steganography.

Frequency domain data hiding: Images are first transformed into frequency domain, and then data are embedded by modifying the transformed coefficients. Frequency domain steganography often suffers from relatively higher computational cost and lower embedding capacity than those of spatial domain data hiding.

Compressed domain data hiding: Data hiding is achieved by modifying the coefficients of the compressed code of a cover image. Since most images transmitted over Internet are in compressed format, embedding secret data into the compressed domain would arouse little suspicion.

The important factors needed to consider when we are designing a new information hiding scheme are embedding capacity and visual quality of stego images. But embedding capacity and visual quality are inversely proportional to each other. That is, if embedding capacity is increased, then the visual quality is decreased and vice versa.

The aim of this paper is to present an information hiding method to improve the embedding capacity and to increase the security by using Sudoku. The total number of a classical 9×9 Sudoku solution is around 6.671x$10^{21}$. Chang et al. recognized that these huge solution spaces are indeed provides distinct input keys for image steganography, and thus proposed a new scheme based on Sudoku for data hiding [1]. This paper not only focuses on hiding the information but also on the security of information by providing user input to the Sudoku. Section 2 deals with Literature survey and related work, Section 3 deals with proposed method of data embedding and data extraction followed by conclusion.

1. **RELATED WORK**

Sudoku is the logic-based number placement puzzle whose objective is to fill a 9×9 grid using the digits from 1 to 9. A solution of Sudoku grid satisfies following properties. First, a Sudoku grid contains nine 3×3 sub-blocks, each contains different digits from 1 to 9. Second, each row and each column of a Sudoku grid also contain different digits from 1 to 9. Now we will briefly describe Chang et al. Steganographic scheme based on Sudoku solutions.

The central idea of Chang et al’s [1] method is to modify the selected pixel pairs in the cover image based on a specially designed reference matrix *M* to insert secret digits. For an 8-bit grey scale cover image, the size of the reference matrix *M* is designed to be 256 × 256. To construct a reference matrix *M,* a “Tile” matrix *T* is constructed first by subtracting every digit in a Sudoku puzzle by one, so that the digits in the matrix *T* are ranged from 0 to 8, as shown in the Figure 1(a) and 1(b). The reference matrix *M* is then consisting of an *m* × *m* tiling of copies of *T*,

 Where:

 *m* = ⎣$\frac{256}{9}$⎦+1

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 7 | 8 | 2 | 4 | 9 | 1 | 5 | 3 | 6 |
| 1 | 4 | 6 | 5 | 7 | 3 | 9 | 2 | 8 |
| 5 | 3 | 9 | 6 | 2 | 8 | 7 | 4 | 1 |
| 3 | 5 | 8 | 4 | 6 | 4 | 2 | 9 | 7 |
| 4 | 9 | 1 | 5 | 5 | 2 | 8 | 6 | 3 |
| 6 | 2 | 7 | 6 | 8 | 9 | 4 | 1 | 5 |
| 2 | 7 | 5 | 9 | 3 | 6 | 1 | 8 | 4 |
| 8 | 1 | 3 | 2 | 4 | 5 | 6 | 7 | 9 |
| 9 | 6 | 4 | 8 | 1 | 7 | 3 | 5 | 2 |

 Figure 1(a): A Sudoku Solution

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 6 | 7 | 1 | 3 | 8 | 0 | 4 | 2 | 5 |
| 0 | 3 | 5 | 4 | 6 | 2 | 8 | 1 | 7 |
| 4 | 2 | 8 | 5 | 1 | 7 | 6 | 3 | 0 |
| 2 | 4 | 7 | 0 | 5 | 3 | 1 | 8 | 6 |
| 3 | 8 | 0 | 6 | 4 | 1 | 7 | 5 | 2 |
| 5 | 1 | 6 | 2 | 7 | 8 | 3 | 0 | 4 |
| 1 | 6 | 4 | 8 | 2 | 5 | 0 | 7 | 3 |
| 7 | 0 | 2 | 1 | 3 | 4 | 5 | 6 | 8 |
| 8 | 5 | 3 | 7 | 0 | 6 | 2 | 4 | 1 |

 Figure 1(b): A “Tile” Matrix (*T*)

Note that matrix *M* has to be truncated to 256 × 256. An example of a reference matrix *M* for a given solution of a Sudoku puzzle is shown in Figure 2.

* **Chang’s method**
* **Data Embedding**

In the data embedding phase, Chang et al. first converted the secret bit stream into secret digits in the base-9 numeral system, and then embed these secret digits into the cover image. Suppose the converted secret digits are denoted by *S* =$ s\_{1}s\_{2}s\_{3}……s\_{n}$, where *n* is the total number of converted secret digits and $s\_{k}$∈ [0, 8], 1 ≤ *k* ≤ *n*. In order to embed secrete digits, the cover image is partitioned into *R* non-overlapping blocks of size 1 × 2 , and $i^{th}$ block consists of a pixel pair ($p\_{i1} , p\_{i2}$).



 Figure 2: Reference Matrix

To embed a secret digit $s\_{i}$into pixel pair ($p\_{i1} , p\_{i2}$), first the element $P\_{i}$=*M* [$p\_{i1} , p\_{i2}$] is located. Once the element $P\_{i} $has been located, three sets of candidate elements [1]$ CE\_{H}$, $CE\_{V}$and $CE\_{B}$can be constructed according to the location of$ P\_{i}$*.* Here,$ CE\_{H}$, $CE\_{V}$and $CE\_{B}$ respectively represents the sets of horizontal, vertical and boxed candidate elements, as depicted in Figure 3. Note that the digits in each set consist numbers from 0 to 8.

To conceal a base-9 secret digit $s\_{i}$ into block *i* with pixel pair ($p\_{i1} , p\_{i2}$), three candidates *M* [$x\_{H} , y\_{H}$], *M* [$x\_{V} , y\_{V }$] and *M* [$x\_{B} , y\_{B}$] are selected respectively from$ CE\_{H}$, $CE\_{V}$and $CE\_{B}$such that

*M* [$x\_{H} , y\_{H}$] = *M* [$x\_{V} , y\_{V }$] = *M* [$x\_{B} , y\_{B}$] = $s\_{i}$

The embedding is done by modifying pixel pair ($p\_{i1} , p\_{i2}$) to$ (p\_{i1}^{'}$,$ p\_{i2}^{'}$) based on the minimum Manhattan distance between ($p\_{i1} , p\_{i2}$) and these three candidates, i.e.,

$min\_{j=H,V,B} \{ $| ($p\_{i1}- x\_{j }$| + | ($p\_{i2}- y\_{j }$| }

Once the candidate pair ($x\_{min} , y\_{min}$) with minimum Manhattan distance has been obtained, the cover pixel pair ($p\_{i1} , p\_{i2}$) is then modified to ($x\_{min} , y\_{min}$) for concealing a secret digit $s\_{i}$, i.e.,

$ (p\_{i1}^{'}$,$ p\_{i2}^{'}$) ← ($x\_{min} , y\_{min}$).



Figure 3: Illustration of $CE\_{H}$(solid line), $CE\_{B}$(dashed line),$ CE\_{V}$(dotted line). *M* [$p\_{i1}$,$ p\_{i2}$]= *M* [12, 5]= 3.

* **Data Extraction**

The embedded data can be extracted directly from the stego image by referencing the same Sudoku solution used in the embedding phase. To extract secret digits, stego image is partitioned into no overlapping blocks of 1× 2 pixels using the same technique used in embedding phase. For block *i* with pixel pair$ (p\_{i1}^{'}$,$ p\_{i2}^{'}$), the secret digit is simply obtained by referencing the reference matrix *M* at the position$ (p\_{i1}^{'}$,$ p\_{i2}^{'}$), i.e., *M*$ (p\_{i1}^{'}$,$ p\_{i2}^{'}$) = $s\_{i}$.

* **Hong’s method**

This is based on Chang’s method. It was found that more suitable pairs in the reference matrix *M* cause smaller image distortion but, these pairs are not included in the sets of candidate pairs used in Chang et al.’s method. Therefore, Hong [2] uses an additional set of candidate elements $ CE\_{A}$ [2]is defined by

$ CE\_{A}$= {$\left(a\_{i1},a\_{i2}\right)|\left(a\_{i1},a\_{i2}\right) ∩( CE\_{H}$ $∪CE\_{V}∪CE\_{B})= ∅$

$D\_{m}$(($p\_{i1} , p\_{i2}$),$ \left(a\_{i1},a\_{i2}\right)$)<4}

i.e., $ CE\_{A}$is a set of pairs that disjoint the pairs defined in$ CE\_{H}$, $CE\_{V}$and$ CE\_{B}$, and their Manhattan distances to the located pair are smaller than 4. Now along with the candidate element obtained from Chang et al.’s method additional elements have to be considered while embedding [1].

* **Using Color Pixels for Embedding**

Both Chang et al.’s and Hong et al.’s method use grey scale image for hiding the data, But Roshan Shetty et al.’s method use color image [3].

Instead of using 2 pixels as in grey scale image a color image has 24 bits in it, 8-bit each for Red (*R*), Green (*G*) and Blue (*B*) components respectively. So, *R* and *G* component of one pixel is enough to guide the cover pixel for embedding the data.

1. **PROPOSED METHOD**

In this section we shall propose a new version of data hiding scheme using Sudoku by combining the methods proposed by Chang et al., Hong et al. and Roshan Shetty et al.’s methods, and also we make sure that half of the values to the Sudoku puzzle is entered by the sender and not fully computer generated.

* **Data Embedding**

Any image onto which secret information has to be embedded is chosen. Each pixel of this image is extracted and 2 components of pixel: Red (*R*) and Green (*G*) color are chosen for embedding. Each color component is an 8-bit binary number. The 8-bits represent numbers ranging from 0 to 255. *R* and *G* are chosen as x-axis and y-axis components of reference matrix *M*, forming pair ( $p\_{i1}, p\_{i2}$)

, where $p\_{i1}$ = *R* and $p\_{i2}$= *G*. Then 3 candidate elements are chosen called horizontal ($CE\_{H }), $vertical ($CE\_{V })$ and boxed ($CE\_{B})$ as shown in the Figure 3. Now these candidate elements are determined as follows:

S1: ( i.e., select the candidate element for $CE\_{H}$)

If $ p\_{i2}$>3 and$ p\_{i2}$<252,

Then$ CE\_{H }$= {*M* ($p\_{i1}, p\_{i2}$ – 4), *M* ($p\_{i1}, p\_{i2}$ – 3), *M* ($p\_{i1}, p\_{i2}$ – 2), *M* ($p\_{i1}, p\_{i2}$ – 1), *M* ($p\_{i1}, p\_{i2}$), *M* ($p\_{i1}, p\_{i2}$ +1), *M*($p\_{i1}, p\_{i2}$ +2), *M* ($p\_{i1}, p\_{i2}$ +3), *M* ($p\_{i1}, p\_{i2}$ +4)};

Else If $p\_{i2}\leq $ 3,

Then $CE\_{H }$= {*M* ($p\_{i1}$, 0), *M* ($p\_{i1}$, 1), *M* ($p\_{i1}$, 2), *M* ($p\_{i1}$, 3), *M* ($p\_{i1}$,4), *M* ($p\_{i1}$, 5), *M* ($p\_{i1}$, 6), *M* ($p\_{i1}$, 7), *M* ($p\_{i1}$, 8)};

Else If $p\_{i2}\geq $ 252,$ $

Then $CE\_{H }$= {*M* ($p\_{i1}$, 247), *M* ($p\_{i1}$, 248), *M* ($p\_{i1}$, 249), *M* ($p\_{i1}$, 250), *M* ($p\_{i1}$, 251), *M* ($p\_{i1}$, 252), *M* ($p\_{i1}$, 253), *M* ($p\_{i1}$, 254), *M* ($p\_{i1}$, 255)}.

S2 :( i.e., select the candidate element for$ CE\_{V }$)

If $ p\_{i1}$>3 and$ p\_{i1}$<252,

Then$ CE\_{V }$={*M*($p\_{i1}– 4, p\_{i2}$)},*M*($p\_{i1}– 3, p\_{i2}$), *M*($p\_{i1}– 2, p\_{i2}$) , *M*($p\_{i1}– 1, p\_{i2}$), *M*( $p\_{i1}, p\_{i2}$), *M*( $p\_{i1} +1, p\_{i2}$), *M*( $p\_{i1}+2, p\_{i2}$), *M*( $p\_{i1}+3, p\_{i2}$), *M*( $p\_{i1}+4, p\_{i2}$)};

Else If $p\_{i1}\leq $ 3,

Then $CE\_{V }$= {*M* ($ 0,p\_{i2}$), *M* ($ 1,p\_{i2}$), *M* ($ 2,p\_{i2}$), *M*($ 3,p\_{i2}$), *M*($ 4,p\_{i2}$), *M*($ 5,p\_{i2}$), *M*($ 6,p\_{i2}$), *M*($ 7,p\_{i2}$), *M*($8 ,p\_{i2}$)};

Else If $p\_{i1}\geq $ 252,$ $

Then $CE\_{V }$= {*M* ($ 247,p\_{i2}$), *M* ($ 248,p\_{i2}$), *M* ($ 249,p\_{i2}$), *M*($ 250,p\_{i2}$), *M*($ 251,p\_{i2}$), *M*($ 252,p\_{i2}$), *M* ($ 253,p\_{i2}$), *M*($ 254,p\_{i2}$), *M*($255 ,p\_{i2}$)};

S3: (i.e., select the candidate element for $CE\_{B }$)

If $p\_{i1}$ < 252 and $p\_{i2}$ < 255,

Then compute $x\_{B}$= $\left⌊\frac{p\_{i1}}{3}\right⌋$ × 3, $ y\_{B}$= $\left⌊\frac{p\_{i2}}{3}\right⌋$ × 3, and $CE\_{B}$={*M*( $x\_{B}, y\_{B}$), *M* ($x\_{B}, y\_{B}+1$), *M*( $x\_{B}, y\_{B}$+2),*M*( $x\_{B}+1,y\_{B}$), *M*($x\_{B}+1, y\_{B}+1$), *M*( $x\_{B} +1, y\_{B}+2$), *M*( $x\_{B}+2, y\_{B}$), *M*( $x\_{B}+2, y\_{B}+1$), *M*( $x\_{B}+2, y\_{B}+2$)};

Else$ CE\_{B }= <EMPTY>$.

$ CE\_{A}$= {$\left(a\_{i1},a\_{i2}\right)|\left(a\_{i1},a\_{i2}\right) ∩( CE\_{H}$ $∪CE\_{V}∪CE\_{B})= ∅$

$D\_{m}$(($p\_{i1} , p\_{i2}$),$ \left(a\_{i1},a\_{i2}\right)$) < 4}

According to an input secret digit$ s\_{i}$, 4 candidate elements as shown in the Figure 4, *M* ( $x\_{H} , y\_{H}$ ), *M* ( $x\_{V} , y\_{V}$ ), *M* ( $x\_{B} , y\_{B}$), *M* ( $x\_{A} , y\_{A}$ ), are found from $CE\_{H}$,$ CE\_{V}$,$ CE\_{B}$,$ CE\_{A}$ respectively. In other words if $p\_{i1} and p\_{i2}$ are smaller than 255, then the found candidate element satisfy *M*($x\_{H} , y\_{H}$) = *M*($x\_{V} , y\_{V}$) = *M*($x\_{B} , y\_{B}$) = *M*($x\_{A} , y\_{A}$) = $s\_{i}$; otherwise

*M*($x\_{H} , y\_{H}$) = *M*($x\_{V} , y\_{V}$) = *M*( $x\_{A} , y\_{A}$ ) = $s\_{i}$. The cover pixel pair ( $p\_{i1} , p\_{i2}$ ) is modified as $(p\_{i1}^{'}$,$ p\_{i2}^{'}$) by a minimum distortion candidate element *M* ($x\_{min} , y\_{min})$ which is selected by

*M* ($x\_{min} , y\_{min}$) = $min\_{j = H,V,B,A} \{$ | ($p\_{i1}- x\_{j })$ | + | ($p\_{i2}- y\_{j })$ | }.

Thus, the cover pixel pair ($p\_{i1} , p\_{i2}$) is modified as$ (p\_{i1}^{'}$,$ p\_{i2}^{'}$) to conceal the secret digit $s\_{i} $with small distortions. This small distortion does not affect the image much and no physical difference can be identified of this image with respect to original image. After performing the above replacement for all digits of secret information the embedding phase is complete.

The main requirement for this paper is that the Sudoku solution has to be same at both sender and receiver end. In our method Sudoku solution is half entered by the user, this provides more security to the secret message as it takes more time for the receiver to solve the puzzle and get the information. Backtracking Algorithm is used to solve the Sudoku puzzle. The size of secret information is embedded initially on the cover image so that the receiver can determine how many pixel to scan and retrieve data. Since digital media can be any type of data such as text, image, audio, etc.; it is necessary for the receiver to be aware of it. This is done by embedding them after embedding secret digit on the cover image itself (Figure 5). At last the extension itself is embedded.



Figure 4:$CE\_{H}$, $CE\_{V}$, $CE\_{B}$, $CE\_{A}$( dark shaded region)

 Size embedded

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

 Secret message Extension embedded

 Figure 5: General format of embedding data

* **Data Extraction**

The embedded secret digits can be exactly extracted from the received stego image with the same Sudoku solution used in embedding phase. In this phase first each pixel is extracted from stego image then from this Red (*R*) and Green (*G*) components of each pixel are used similar to that in embedding phase.

The incomplete Sudoku puzzle which is shared between the user and the receiver is required to be solved first, by the receiver and then convert it into a reference matrix by subtracting 1 from the Sudoku solution similar to the one shown in Figure 1(b). The *R* and *G* are chosen as x-axis and y-axis components of Sudoku solution forming pair ( $p\_{i1} , p\_{i2}$ ) where $p\_{i1}=$ *R* and $p\_{i2}$= *G*. The value at position$(p\_{i1},p\_{i2}$) is the required secret digit. This process is done for all the pixel and data is extracted. The obtained secret digit which is in base 9 is converted to base 2. This completes the extraction phase.

1. **COMPARATIVE ANALYSIS**

In this section we compare the experimental results obtained from Change et al.’s and Hong et al,’s method. Visual quality of stego image and embedding capacity are two main factors for evaluating the performance of an information hiding system. The peak-signal-to-noise-ratio (PSNR) is applied in simulations to evaluate the visual quality of stego images produced by the information hiding systems. A large PSNR value means that the stego image has good visual quality (i.e., small distortion). On the contrary, a small PSNR value indicates that the stego image has poor visual quality (i.e., large distortion).

For embedding capacity(C), a stego image is to carry secret data as many as possible. The embedding capacity is evaluated by the number of secret bits that can be embedded into a cover pixel. A large value of C indicates that a cover image can be used to hide a large number of secret bits. On the contrary, a small value of C means that an information hiding method cannot conceal too many secret bits into the cover image. The following table shows the results of different methods obtained for the image Lena:

|  |  |  |
| --- | --- | --- |
| Method | PSNR(dB) | Capacity(bpp) |
| Zang and Wang | 52.12 | 1 |
| Mielikainan | 52.77 | 1 |
| Chang | 44.97 | 1.5 |
| Hong | 47.49 | 1.5 |
| Roshan Shetty | 47.57 | 3 |

Table 1: Comparative Analysis

From the above table it is observed that Hong’s method yields high capacity when compared to Chang’s method, and Roshan Shetty’s method yields more capacity with using color pixels compared to Hong. Our method is combination of Hong and Roshan Shetty’s method. So we can conclude that our method will also have an embedding capacity equal to 3 and the image quality will not be less than 47.49. Thus if we combine two methods the result will be encouraging.

1. **CONCLUSION & FUTURE WORK**

In this paper we have reviewed work on steganography based on Sudoku Puzzle. Along with the study we also propose a revised version of data hiding scheme based on Hong et al.’s method and Roshan Shetty et al.’s. In Hong’s method an additional set of candidate elements $CE\_{A}$ is used for referencing, so that the function of the reference matrix *M* can be fully exploited. But he used grey scale image for embedding. Here we use color image hence only 1 pixel is enough to embed the data. The proposed data hiding scheme can be deployed for a wide range of applications such as medical imaging, law enforcement, military applications etc., where high image quality is required since the quality of the image is not affected much before and after embedding.

**REFERENCES**

1. C.C. Chang, Y.C. Chou and T.D. Kieu, An Information Hiding Scheme Using Sudoku, Proceedings of the Third International Conference on Innovative Computing, Information and Control (ICICIC2008), June 2008.
2. Wien Hong, Tung-Shou Chen, Chih-Wei Shiu, Steganography Using Sudoku Revisited Second International Symposium on Intelligent Information Technology Application 978-0-7695-3497-8/08, 2008.
3. Roshan Shetty B R, Rohith J, Mukund V, Rohan Honwade Steganography using Sudoku Puzzle, 2009 International Conference on Advances in Recent Technologies in Communication and Computing 978-0-7695-3845-7/09, 2009.
4. Sanmitra Ijeri, Shivananda Pujeri, Shrikanth B and Usha B A, Image Steganography using Sudoku Puzzle for Secured Data Transmission, International Journal of Computer Applications (0975 – 888) Volume 48– No.17, June 2012.