DESIGN AND VERIFICATION OF UNIQUE KEY BASED SUDOKU FOR STEGANOGRAPHY IN IMAGES

Ragya Kori, JEC Jabalpur

Abstract: This work intends to give an overview of image steganography, its uses & techniques. Paper work is an implementation of Image Steganography of same plaintext, the implementation of a system that design of unique Sudoku generation for Sudoku based data hiding and high security is the main concern of this paper. The system is based on a hybrid algorithm that applies the techniques of Sudoku and steganography to offer different security features to images transmitted between entities in internet. Based on the proposed algorithm, the authenticity and integrity of the transmitted images can be verified either in the spatial domain or in the encrypted domain or in both domains. The work is implemented on MATLAB design and simulation tool. Paper work uses modified Sudoku based data protection and SNR is very high in present paper work.

Keywords: Peak Signal to Noise Ratio (PSNR), Mean Square Error (MSE), Bit Per Pixel (BPP)

I-INTRODUCTION

The paper work is the Implementation for steganography tools for hiding more data or information which may be text or another image files. Transform techniques will be used for identifying appropriate data. Available methods in area for steganography are as used as following:-

- Initially LSB substitution in image was popular [15],[16]
- Different schemes for LBS substitution developed [14]
- Key based LSB substitution developed [13]
- Multiple layer for steganography developed like steganography with steganography [12]
- DCT based image analysis for finding out area where data may be hiding appropriately [3]
- DWT based image analysis for finding out area where data may be hiding appropriately & found better than DCT based image analysis [7][6]
- Sudoku based data hiding method in cover image [1],[2]

II-METHODOLOGY

Proposed design resolve the problems of available work with significant modification done in interpolation method and Sudoku generation method, one 8 digit public and one 8 digit privet key are using for the proposed work and association of both 8 digit public and privet key develop one 8 bit digit Sudoku Key. Proposed work is a new design which uses Sudoku developed with 8 digits Sudoku Key provided by user, a unique Sudoku solution develop for each different key, this concept of unique Sudoku and combination of public and privet key makes proposed method robust against external intrusion. Proposed work is also using a interpolation and new pixel generation with help of four surrounding pixels instead of two pixels as was use by Chin-Chen Chang et al [1], four pixels interpolation develops more accurate new pixel then two pixels interpolation. The remaining processes of hiding are similar as use by Chin-Chen Chang et al [1].

2.1 DATA HIDING METHOD ADOPTED:

Figure 1 below explains the flow diagram of the proposed work, it can be explains as in following steps

Step1: Input an 8 digit decimal public key, the key can be any value of 8 digit and must be pass when data hiding required.

Step2: perform logical XOR between 8 digit public key and 8 digit privet key which was already in the steganography module. The output key is known as Sudoku Key.

Step 3: develop a unique t1 to t9 digits out of 8 digits Sudoku key, the t1 to t9 can be anything between 1 to 9 but all t1 to t9 will be different from each other.

Step 4: Assign the values of t1 to t9 into a fixed bottom up approach Sudoku problem. Total 23 unknown values of Sudoku problem will be assigning using t1 to t9 digits.

Step 5: Solve the bottom up Sudoku problem with using Sudoku Rules, proposed method use the method of solving Sudoku by Ahmed Abdulkarim Almuhrij et al [27] from University of Manchester.

.

 ©2017 IJAIR. All Rights Reserved

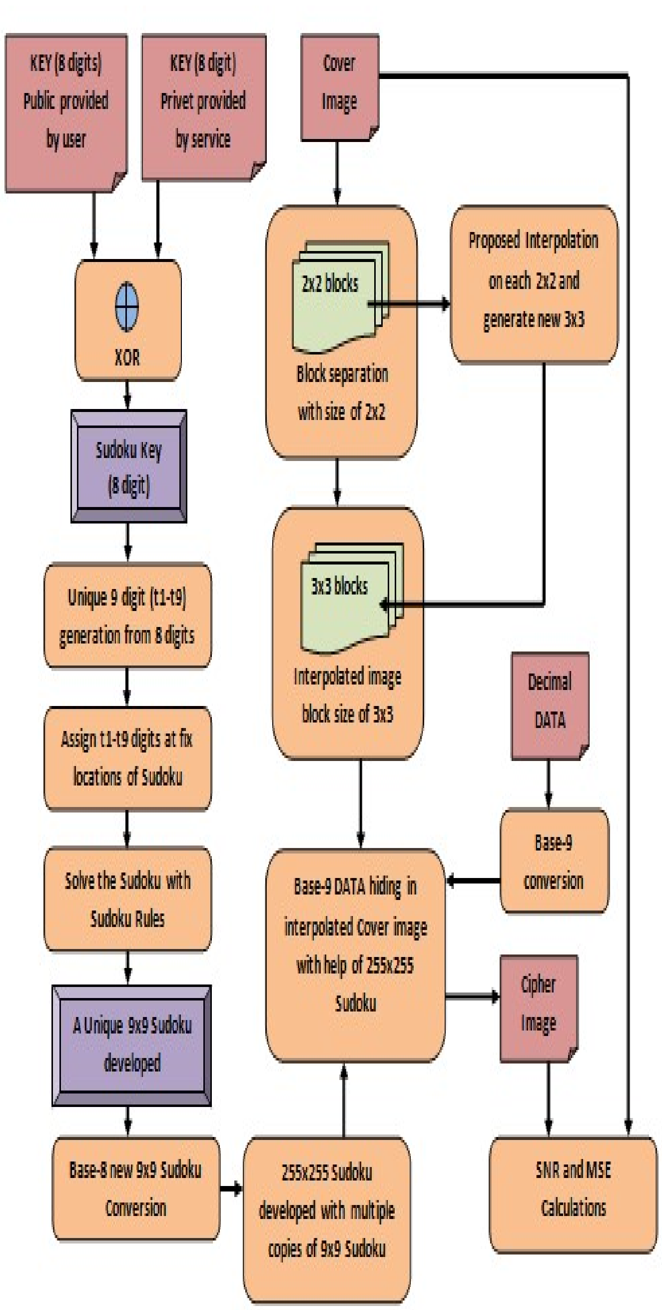

Figure 1 Data hiding Flow diagram

Step 6: convert the base-9 Sudoku into base-8 Sudoku because we can hide digits 0-8 only.
Step 7: develop a 255x255 Sudoku matrix with help of base-8 developed Sudoku, the size of 255x255 is fixed because any pixel of image cannot be greater than 255.
Step 8: input the cover image and isolate its 2x2 blocks
Step 9: perform interpolation on each 2x2 block and develop new 3x3 interpolated block, total 4 pixels was available in 2x2 block and total 9 pixels develop in 3x3 block, means 5 new pixels developed. These 5 new pixels can be modified according to the data which is to be hidden in the cover image.
The roll of interpolation is that original pixels (4 pixel of 2x2 block) must not be change and new 5 interpolated pixels can be modified with small amount, hence overall image will modified less as required in steganography.
Step 10: input the data which can be character or any number first convert it into its ASCII values then develop a single string of decimal digits.
Step 11: convert the number into base-9
Step 12: start hiding the data in cover image with help of 255x255 Sudoku matrix. Actually the original digit is not hiding in the cover image, the digits of data simply modify the interpolated pixels of cover image so small modification done. Each 2x2 block of interpolated image can hide 3 digits

2.2 PROPOSED ALGORITHM FOR DATA HIDING : Input a 8 digit public key its 8 digits are K1, K2….K8
KEY=K1K2K3K4K5K6K7K8
Shift row for making Key complex as also done in AES
NK=K2K3K4K5K6K7K8K1
Logical XOR NK with privet key 'PK' for public Key encryption
MK=NK xor PK
Now develop sub-keys from MK
MK1=99999999-MK
MK2=88888888-MK
MK3=77777777-MK
MK4=66666666-MK
MK5=55555555-MK
MK6=44444444-MK
MK7=33333333-MK
MK8=22222222-MK
On the other hand Let Sud is a 9x9 matrix where the position of t1,t2….t9 are fixed

$$Sud = \begin{matrix} t1 & UN & UN & UN & UN & 2 & UN & t3 & UN \\ UN & t4 & UN & UN & t5 & UN & UN & U & 6 \\ UN & UN & t7 & t8 & UN & UN & t9 & UN & UN \\ UN & UN & t1 & t2 & UN & UN & t3 & UN & UN \\ UN & 5 & UN & UN & t4 & UN & UN & UN & t7 \\ t6 & UN & UN & UN & UN & t8 & UN & UN & UN \\ t9 & UN & UN & UN & UN & UN & UN & t1 & UN \\ UN & t2 & UN & UN & UN & UN & UN & UN & t3 \\ UN & UN & t4 & UN & UN & UN & t5 & UN & UN \end{matrix}$$

 ©2017 IJAIR. All Rights Reserved

From the Sub-Keys following method will generate t1, t2…..t9, the values of t1, t2 ….t9 can never be same.
Initially i=1 and j=1 and p=1
'i' can be range maximum from 1 to 23
'j' can be range maximum from 1 to 9
'p' can be range maximum from 1 to 8

$$For\ i = 1:23$$
$$For\ j = 1:8$$
$$if\ \left(MK(j)_{dp} \neq (t_{1:i-1}, t_{1:i-2}, t_{1:i-3} \ldots\ldots t_1\right)$$
$$t_i = MK(j)_{dp}$$
$$elseif\ \left(MK(j)_{d(p+1)} \neq (t_{1:i-1}, t_{1:i-2}, t_{1:i-3} \ldots\ldots t_1\right)$$
$$t_i = MK(j)_{d(p+1)}$$
$$elseif\ \left(MK(j)_{d(p+2)} \neq (t_{1:i-1}, t_{1:i-2}, t_{1:i-3} \ldots\ldots t_1\right)$$
$$t_i = MK(j)_{d(p+2)}$$
$$\vdots$$
$$\vdots$$
$$else$$
$$t_i = MK(j)_{d(p+7)}$$
$$end$$
$$end$$

Ones the value of t1 to t9 computed, it will be according to public key provided by user and privet key by the service provider, a unique Sudoku need to be design and according to computed values of t1 to t9.

The values of UN in Sudoku matrix 'sud' will be computed using Sudoku solver according the Sudoku rules

- Any row cannot have repeated numbers 1 to 9
- Any column cannot have repeated numbers 1 to 9
- Any 3x3 square (corning and central)matrix numbers cannot have repeated numbers 1 to 9

A MATLAB define function names SudokuSolver.m can solve this Sudoku and it can compute all the unknown using above rules, the complete Sudoku is represented below where t1-t9 initially computed with the help of the KEY and U1-U59 computed with the help of SodokuSolver and Sudoku rules.

This Sudoku is completlay define by the key as the key changes the complete Sudoku will also change, and this method will provide a good avalanche which is small change in input KEY cause major changes in output Sudoku, hence the system can be consider a chaotic system.

$$Sud1 = \begin{matrix} t1 & U1 & U2 & U3 & U4 & t2 & U5 & t3 & U6 \\ U7 & t4 & U8 & U9 & t5 & U10 & U11 & U12 & t6 \\ U13 & U14 & t7 & t8 & U15 & U16 & t9 & U17 & U18 \\ U19 & U20 & t10 & t11 & U21 & U22 & t12 & U23 & U24 \\ U25 & t13 & U26 & U27 & t14 & U28 & U29 & U30 & t15 \\ t16 & U31 & U32 & U33 & U34 & t17 & U35 & U36 & U37 \\ t18 & U38 & U39 & U40 & U41 & U42 & U43 & t19 & U44 \\ U45 & t20 & U46 & U47 & U48 & U49 & U50 & U51 & t21 \\ U52 & U53 & t22 & U54 & U55 & U56 & t23 & U57 & U58 \end{matrix}$$

'SudM' will contains digits from 0 to 8 only can be develop as: SudM=Sud1 -1

Next make copies of SudM such that a matrix of 255x255 generated

SudM SudM ……………..SudM
SudM SudM ……………..SudM
.. ..
.. ..
.. ..
SudM SudM ……………..SudM

Figure 2: 255x255 Sudoku matrix

Input the cover image and convert it into matrix form which shows the pixels of the image. Let one part of the cover image as below

$$img = \begin{matrix} r1_1 & r1_2 & r1_3 \\ r2_1 & r2_2 & r2_3 \\ r3_1 & r3_2 & r3_3 \end{matrix}$$

Perform interpolation using the formula below

$$\boldsymbol{rx_{ni}} = \left[\boldsymbol{rx_i} + \left(\frac{\boldsymbol{rx_i + rx_{i+1}}}{\boldsymbol{2}}\right)\right] \Big/ \boldsymbol{2} \qquad \textbf{\textit{where x is constant}}$$

 ©2017 IJAIR. All Rights Reserved

$$cx_{ni} = \left[ri_x + \left(\frac{ri_x + r(i+1)_x}{2}\right)\right] \Big/ 2 \qquad \textit{where } x \textit{ is constant}$$

$$rcx_{ni} = \left\{\left[rx_{ni} + \left(\frac{rx_{ni} + rx_{n(i+1)}}{2}\right)\right] \Big/ 2 + \left[cx_{ni} + \left(\frac{cx_{ni} + cx_{n(i+1)}}{2}\right)\right] \Big/ 2\right\} \Big/ 2$$

$$img1 = \begin{matrix} r1_1 & r1_{n1} & r1_2 & r1_{n2} & r1_3 & r1_{n3} \\ c1_{n1} & rc1_{n1} & c1_{n2} & rc1_{n2} & c1_{n3} & rc1_{n3} \\ r2_1 & r2_{n1} & r2_2 & r2_{n2} & r2_3 & r2_{n3} \\ c2_{n1} & rc2_{n1} & c2_{n2} & rc2_{n2} & c2_{n3} & rc2_{n3} \\ r3_1 & r3_{n1} & r3_2 & r3_{n2} & r3_3 & r3_{n3} \\ c3_{n1} & rc3_{n1} & c3_{n2} & rc3_{n2} & c3_{n3} & rc3_{n3} \end{matrix}$$

Input data 'D' and convert it into ASCCII format and convert the data into base-9 formats

$ND=(D)_9$

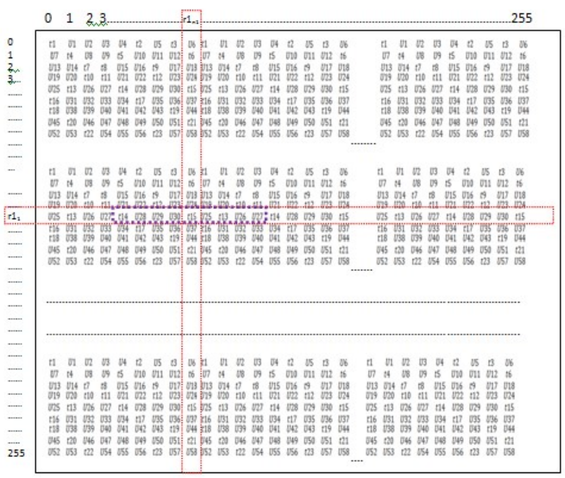

Figure 3 data in cover image modified according this 255x255 Sudoku matrix

Now from the interpolated cover image 'img1' select $(r1_1, r1_{n1})$ position from Sudoku obtain at step 8 and then consider 4 pixel ahead and 4 pixels from back from the position $(r1_1, r1_{n1})$

The nine digits of Sudoku [t14 U28 U29 U30 t15 U25 t13 U26 U27] search for the first digit of new data 'ND(1)' and detect its position $(r1_1, Y_{n1})$

Replace the searched new position $(r1_1, Y_{n1})$ with pixel of $(r1_1, r1_{n1})$ of interpolated image, this process keep on until all the digits of the new data (ND) are not replace,

$$g1 = \begin{matrix} r1_1 & Y_{n1} & r1_2 & Y_{n2} & r1_3 & Y_{n3} \\ Y_{n4} & rc1_{n1} & Y_{n5} & rc1_{n2} & Y_{n6} & rc1_{n3} \\ r2_1 & Y_{n7} & r2_2 & Y_{n8} & r2_3 & Y_{n9} \\ Y_{n10} & rc2_{n1} & Y_{n11} & rc2_{n2} & Y_{n1} & rc2_{n3} \\ r3_1 & Y_{n13} & r3_2 & Y_{n14} & r3_3 & Y_{n15} \\ Y_{n16} & rc3_{n1} & Y_{n17} & rc3_{n2} & Y_{n18} & rc3_{n3} \end{matrix}$$

$\textit{where } x \textit{ is constant}$

The whole idea is that we are not making any change in the original information pixels of the original cover image, the pixel values of interpolated pixel gets modified which was generated using interpolation formula, and making slight change in this pixel does not affect quality on the image significantly.

2.3PROPOSED METHOD FOR DATA EXTRACTION

Step1: Input an 8 digit decimal public key, the key can be any value of 8 digit and must be pass when data extraction required and it must be same as the time of data hiding.

Step2: perform logical XOR between 8 digit public key and 8 digit privet key which was already in the steganography module. The output key is known as Sudoku Key.

Step 3: develop a unique t1 to t9 digits out of 8 digits Sudoku key, the t1 to t9 can be anything between 1 to 9 but all t1 to t9 will be different from each other.

Step 4: Assign the values of t1 to t9 into a fixed bottom up approach Sudoku problem. Total 23 unknown values of Sudoku problem will be assigning using t1 to t9 digits.

Step 5: Solve the bottom up Sudoku problem with using Sudoku Rules, proposed method use the method of solving Sudoku by Ahmed Abdulkarim Almuhrij et al [27] from University of Manchester.

Step 6: convert the base-9 Sudoku into base-8 Sudoku because we can hide digits 0-8 only.

Step 7: develop a 255x255 Sudoku matrix with help of base-8 developed Sudoku, the size of 255x255 is fixed because any pixel of image cannot be greater than 255.

Step 8: Input the cipher image and isolate its 2x2 blocks

 ©2017 IJAIR. All Rights Reserved

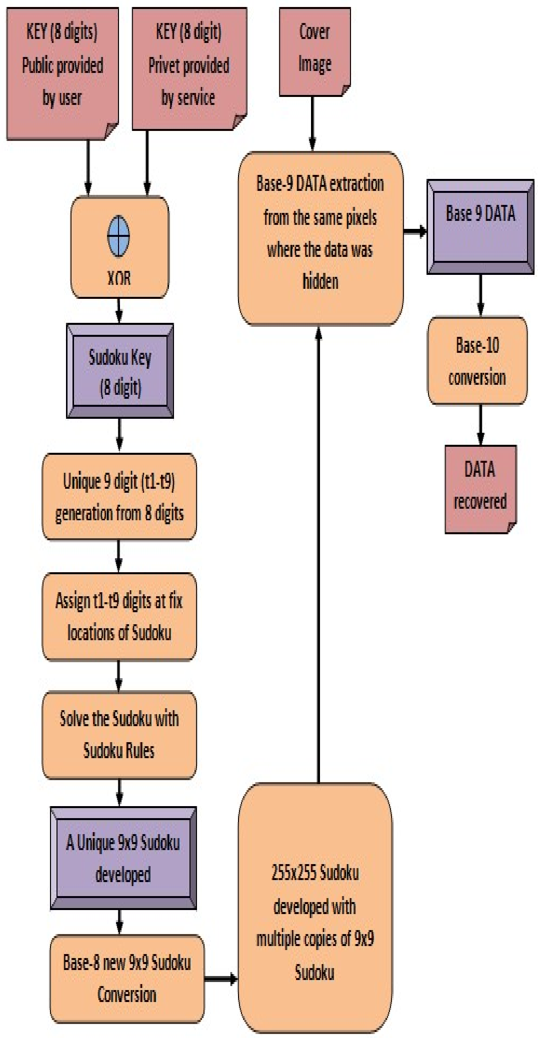

Figure 4 Data Extraction flow diagram

Step 9: With the help of 255x255 Sudoku matrix generate 3 decimal digits from each 2x2 block. The digit must be put as a sting of data.
Step 10: convert the number into base-10
Step 11: develop ASCII from the digit string and then convert ASCII into Characters.

2.4 ALGORITHM FOR DATA EXTRACTION

The process of 255x255 Sudoku will same as was in data hiding
Input the cover image and convert into matrix form which shows the pixels of the image. Let one part of the cover image as below:-

$$img = \begin{bmatrix} P11 & P12 & \cdots & P1(N-1) & P1N \\ P21 & P22 & \cdots & P2(N-1) & P2N \\ P31 & P32 & & P3(N-1) & P3N \\ P41 & P42 & \cdots & P4(N-1) & P4N \\ \vdots & & & & \vdots \\ P(N-1)1 & P(N-1)2 & & P(N-1)(N-1) & P(N-1)N \\ PN1 & PN2 & \cdots & PN(N-1) & PNN \end{bmatrix}$$

Isolate 2x2 block let say

$$Blk = \begin{matrix} P11 & P12 \\ P21 & P22 \end{matrix}$$

Now from the cipher image 'img1' select $(P11, P12)$, $(P11, P21)$ and $(P11, P22)$ position and search corresponding value in 255x255 Sudoku, the values in Sudoku will be data in base-9 form.

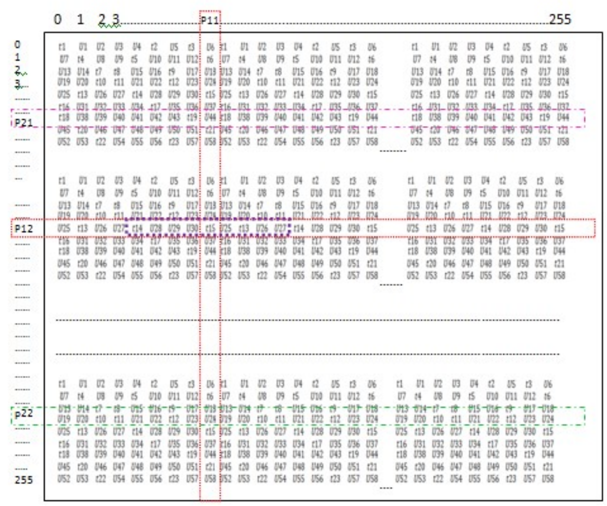

Figure 5: date digits will be find in the 255x255 matrix

Collect values from Sudoku according to positions $(P11, P12)$, $(P11, P21)$ and $(P11, P22)$ and develop a string of base-9 data values
Convert base-9 data string into decimal data string
Convert decimal data string into ASCII and then convert ASCII into alphanumeric characters, these characters will be the final extracted DATA.

III-RESULTS

MSE, SNR and BPP are the result parameters, this parameters will help with observation of results and comparison of proposed work other.

 ©2017 IJAIR. All Rights Reserved

$$MSE = \frac{\{\sum_{i=1}^{rw}\sum_{j=1}^{cl}[P_{ciphe}\ (i,j) - P_{cover}(i,j)]\}^2}{rw * cl}$$

$$PSNR = 10 * log_{10} * \frac{256^2}{MSE}$$

$$BPP = \frac{8 * Bd}{Bi}$$

P_{cipher} is the pixel of cipher image
P_{cover} is pixel of cover image
Rw is number of row in cipher image
Cl is the number of column in cipher image
Bi number of bits in the cipher image
Bd number of bits in data

BPP: Bit Per Pixel is the number of bits that can be hidden inside a pixel, The capacity of data which can be hidden inside the proposed work can be explained with an test image let say image of 'Lena' with 512x512 pixels
512x512=2, 62, 144 pixels
Total 256x256 number of 2x2(4 pixel block) can be develop with 512x512 image
256x256x2x2=2,62,144 pixels
And after interpolation 2x2 block convert into 3x3 block means total
256x256x3x3=5,89,824 pixels
And as it's a color image with 3 frames total pixels can be
256x256x3x3x3= 17,69,472 pixels
As proposed method can hide three digits (12 bit) in each 2x2 block of cover image
17,69,472/4 = 4,42,386 total 2x2 blocks available
And each block can hide 12 bits hence 4,42,386 x 12 = 53,08,632 bits can be hidden in interpolated image size of 256x256x3x3x3 pixels and each pixel of 8 bit hence 256x256x3x3x3x8= 1,41,55,776 bits in interpolated image
BPP= 53,08,632 x 8 / 141,55,776 = 3.000122
Hence the maximum BPP observe for the proposed method is 3.000122

Number of bits	**PSNR**	**MSE x 10^{-11}**
2000	54.53	5.41
4000	53.47	6.91
8000	53.01	7.68
16000	52.15	9.37
32000	50.95	12.37
64000	49.84	15.95
128000	48.83	20.13

Table 1 PSNR observation for various size of data

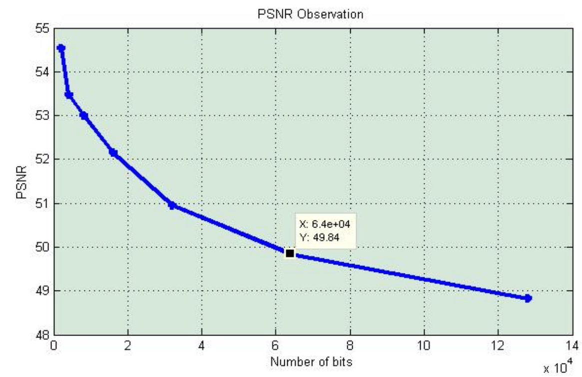

Figure 6 PSNR measure for different numbers of bits

3.1 COMPARATIVE RESULTS

PSNR Results observe for the 512x512 Lena Standard image			
2000 bits of data		**1,20,000 bits of data**	
Chang-Tsun Li et al [13]	**Proposed Work**	**Thai-Son Nguyen et al [2]**	**Proposed Work**
46.25	54.53	48.67	48.83

Table 2 PSNR Comparison with available work

Sr. NO.	**AUTHOR**	**Results**
1.	Fan Li et al [3]	PSNR observed is 52.99 when BPP taken as 0.063 for 512x512 Lena image

 ©2017 IJAIR. All Rights Reserved

2.	Chin-Chen Chang et al [1]	2.23 bpp hidden in Lena Image obtain PSNR of 26.86
3.	Proposed work	3.000122 BPP observe for Lena image of 512x512 pixels

Table 3 BPP Comparison with available work

From the comparative results it can be observe that proposed work PSNR is better than available works and also the proposed work can hide more bits per pixel then available work.

IV-CONCLUSION

The original objective for paper work was to develop an optimized technique for hiding data inside cover image also to reduce amount for data in the channel while stenograph data transmission which is been achieved. A new 8 digit decimal number based unique Sudoku developed for enhancing the robustness of the work and also a modified interpolation is been developed for maintain good quality of image after hiding the data. problem with steganography is that it need lots of data (image) for sending few small amount for data, so proposed work is a good solution for this problem it can be confidently say that because we have achieved very good PSNR proposed design can be use for secure communication in cloud services also. The data bits are embedded in random into the cover-image pixels instead of it sequentially can be done to improve the security of the system in future.

REFERENCES

[1] Chin-Chen Chang, Thai-Son Nguyen, Yanjun Liu, A reversible data hiding scheme for image interpolation based on reference matrix, 2017 5th International Workshop on Biometrics and Forensics (IWBF) , DOI: 10.1109/IWBF.2017.7935098, Coventry, UK IEEE Xplore: 29 May 2017

[2] Thai-Son Nguyen , Chin-Chen Chang, A reversible data hiding scheme based on the Sudoku technique, http://dx.doi.org/10.1016/j.displa.2015.10.003-0141-9382/2015 Elsevier, Science Direct Displays 39 (2015) 109–116

[3] Fan Li, Qian Mao, and Chin-Chen Chang, A Reversible Data Hiding Scheme Based on IWT and the Sudoku Method, International Journal of Network Security, Vol.18, No.3, PP.410-419, May 2016 410

[4] Suman Chakraborty, Prof. Samir K. Bandyopadhyay, Steganography Method Based On Data Embedding By Sudoku Solution Matrix, International Journal of Engineering Science Invention ISSN (Online): 2319 – 6734, ISSN (Print): 2319 – 6726 www.ijesi.org Volume 2 Issue 7, July 2013, PP.36-42

[5] Ruth-Nick-Louise, Creating Sudoku Puzzles, Control # 2883, February 19, 2008 Page 1 of 15 Control # 2883, math.swansea.ac.uk/staff/jhg/papers/MCM2008-Ruth-Nick-Louise.pdf

[6] Sanmitra Ijeri, Shivananda Pujeri, Shrikant B, 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 31

[7] The Duc Kieu, Zhi-Hui Wang, Chin-Chen Chang, and Ming-Chu Li, A Sudoku Based Wet Paper Hiding Scheme, International Journal of Smart Home, Vol.3, No.2, April, 2009

[8] Ayushi A Symmetric Key Cryptographic Algorithm, 2010 International Journal of Computer Applications (0975 - 8887), Volume 1 – No. 15

[9] Sharvari Dixit, Archana Gaikwad, Snehal Gaikwad, Shweta A.Shanwad, Public Key Cryptography Based Lossless and Reversible Data Hiding in Encrypted Images, Volume 6 Issue No. 4 DOI 10.4010/2016.822, ISSN 2321 3361 © 2016 IJESC

[10] Yun-Te Lin, Chung-Ming Wang, Wei-Sung Chen, Fang-Pang Lin, and Woei Lin, Novel Data Hiding Algorithm for High Dynamic Range Images, IEEE TRANSACTIONS ON MULTIMEDIA, VOL. 19, NO. 1, JANUARY 2017

[11] Ali Al-Haj, Hiba Abdel-Nabi, Digital Image Security Based on Data Hiding and Cryptography, 2017 3rd International Conference on Information Management, 978-1-5090-6306-2/17/2017 IEEE

[12] Shripal Rawal, Advanced Encryption Standard (AES) and It's Working, International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056 Volume: 03 Issue: 08 | Aug-2016

[13] Ching-Chun Chang, Chang-Tsun Li, Reversible Data Hiding in JPEG Images Based on Adjustable Padding, 2017 5th International Workshop on Biometrics and Forensics (IWBF),

 ©2017 IJAIR. All Rights Reserved

DOI: 10.1109/IWBF.2017.7935083, IEEE Xplore: 29 May 2017

[14] Donna Maglott, Jim Ostell, Kim D. Pruitt and Tatiana Tatusova, Entrez Gene: gene-centered information at NCBI, D52–D57 Nucleic Acids Research, 2011, Vol. 39, Database issue Published online 28 November 2010, doi:10.1093/nar/gkq1237

[15] Arnab k. maji, Sudipta roy, Rajat k. pal, A Novel Algorithmic approach for solving Sudoku puzzle in Guessed Free Manner, EUROPEAN ACADEMIC RESEARCH, VOL. I, ISSUE 6/ SEPEMBER 2013 ISSN 2286-4822

[16] Ammad Ul Islam, Faiza Khalid, Mohsin Shah, Zakir Khan, Toqeer Mahmood, Adnan Khan, Usman Ali, Muhammad Naeem, An Improved Image Steganography Technique based on MSB using Bit Differencing, INTECH 2016, 978-1-5090-2000-3/16/2016 IEEE

[17] Matthias Wilhelm, Ivan Martinovic, Ersin Uzun, and Jens B. Schmitt, SUDOKU: Secure and Usable Deployment of Keys on Wireless Sensors, INCCON 2010, 978-1-4244-8915-2/10/2010 IEEE

[18] Baochen SUN Xiwei SUN Yue WU Yilong YIN Gongping YANG, A New Algorithm for Generating Unique-solution Sudoku, Fourth International Conference on Natural Computation, 978-0-7695-3304-9/08-2008 IEEE, DOI: 10.1109/ICNC.2008.788

[19] M. Iwata, K. Miyake, A. Shiozaki, Digital steganography utilizing features of JPEG images, IEICE Trans. Fundam. Electron. Commun. Comput. Sci. E87-A(2004) 929–936.

[20] J. Mielikainen, LSB matching revisited, IEEE Signal Process. Lett. 13 (2006) 285–287.

[21] I.J. Cox, J. Kilian, F.T. Leighton, T. Shamoon, Secure spread spectrum watermarking for multimedia, IEEE Trans. Image Process. 6 (1997) 1673–1687.

[22] J. Tian, Reversible data hiding using difference expansion, IEEE Trans. Circuits Syst. Video Technol. 13 (2003) 890–896.

[23] D.M. Thodi, J.J. Rodriguez, Expansion embedding techniques for reversible watermarking, IEEE Trans. Image Process. 16 (2007) 721–730.

[24] Z. Ni, Y.Q. Shi, N. Ansari, W. Su, Reversible data hiding, IEEE Trans. Circuits Syst. Video Technol. 16 (2006) 354–362.

[25] H. Luo, F.X. Yu, H. Chen, Z.L. Huang, H. Li, P.H. Wang, Reversible data hiding based on block median preservation, Inf. Sci. 181 (2011) 308–328.

[26] Y.C. Li, C.M. Yeh, C.C. Chang, Data hiding based on the similarity between neighboring pixels with reversibility, Digital Signal Process. 20 (2010).

[27] Ahmed bAbdulkarim Almuhrij, SUDOKU GAME, Dr. David Lester, 2011, University of Manchester

 ©2017 IJAIR. All Rights Reserved