Zero Distortion Technique:
An approach to image steganography on color images using strength of Chaotic Sequence

Shivani
Computer Science and Engineering
ABES Engineering College
Ghaziabad, India
shivanisharma2804@gmail.com

Virendra Kumar Yadav
Computer Science and Engineering
ABES Engineering College
Ghaziabad, India
virendrashines@gmail.com

Saumya Batham
Master of Computer Applications
ABES Engineering College
Ghaziabad, India
saumyabatham003@gmail.com

ABSTRACT
Steganography is a powerful technique to hide data, i.e. the existence of the private and sensitive data cannot be pursued by the intruder. Combination of cryptography with steganography gives a powerful impact. It becomes a tedious task for the steganalyst to break the code. There are several techniques for performing steganography, out of which LSB is the widely used technique. Limitations of these techniques are like, it degrades the quality of the cover image, blurring, less amount of data can be hidden. Zero Distortion Technique has been proposed to overcome these limitations. Proposed technique consider the image only as a reference and hides data in the locations of the image. The bits of the cover image remains untouched which provides zero distortion in the cover image chaotic sequence is used to encrypt the elements of location matrix. Experimental results on certain color images gives better results in terms of amount of data which can be hidden i.e. in RGB bands. Proposed technique is efficient as there is no distortion in the cover image.

Categories and Subject Descriptors
Security

General Terms
Algorithms, Performance, Design, Reliability, Experimentation, Security.

Keywords
Location matrix, Zero Distortion, Steganalysis, LSB technique, Cryptography, Chaotic Sequence.

1. INTRODUCTION
In today's era of internet confidentiality become the main concern while sending or receiving data over wireless medium.

ICTCS '14, November 14 - 16 2014, Udaipur, Rajasthan, India
Copyright 2014 ACM 978-1-4503-3216-3/14/11…$15.00
http://dx.doi.org/10.1145/2677855.2677905

Sometimes data belongs to some government organization or private organization and it contains not only confidential but also sensitive information. Information hiding means encapsulation of confidential data in any digital medium i.e. text, image, audio/video and protocol. Information hiding is further partitioned into three broad categories viz steganography, digital watermarking and cryptography [1].All the three fields are very vast areas in itself and has their own features. Presented work focuses on steganography along with cryptography.

Steganography is a mechanism in which data can be hidden, by virtue of which the presence of hidden data cannot be detected by any illegal third party or steganalyst. On the other hand cryptography enables us to mould data, making it unreadable [2].Steganalysis is a mechanism through which one can conceal the presence of hidden data from stego image. Steganography when applied to cover image results stego image i.e. image embedded with hidden data. Hiding of data can be performed in gray, color or in black/white images. While applying steganography the main concern is to have less or zero distortion in the cover image. If distortion is less the applied technique becomes more powerful. The probability of detecting hidden message within image is less if information considered for hiding is less in the cover image [5]. A large number of techniques implement steganography. Several limitations to these techniques are less amount can be hidden, changes in the image can be visible to human eye as well as using histogram, blurring in the cover image and message can be easily extracted etc.

The work has been categorized as follows: Section II contains the description of related works in the area of steganography. Zero Distortion Technique has been discussed in section III. Section IV and V contains the figures and tables of proposed method and conclusion respectively. In last experimental results has been presented on certain color images.

2. RELATED WORK
2.1 LSB Technique
Least Significant Bit is used to hide data because the chance of detection is very less by the human eye [3].The rightmost bit is known as the least significant bit. It is the simplest known technique and can be implemented very easily [6], [7].G. Viji and J. Balamurugan [4] in this paper author has elaborated the technique of LSB substitution. They had firstly encrypted the text and then embedded it in the cover image in the LSB bit after doing segmentation .Segmentation means classifying RGB

separately and then again splitting them into blocks so that a large amount of data can be hidden in the cover image. Encryption on the data is carried out to ensure more security to the data.

2.2 Encryption Technique

Encryption is a technique used for making data unreadable to an eavesdropper and providing confidentiality to the data .It means converting the original text into cipher text, known as encryption basically carried out at sender's end [8]. It can be carried out using asymmetric or symmetric key cryptography [9].Symmetric key encryption uses one secret key for encryption and is a very simple and a very ancient technique. Algorithms that uses symmetric key encryption for stream cipher are FISH , SNOW , RC4 , Py , QUAD etc and for block cipher are BLOWFISH , Serpent , AES , DES etc . It is also termed as private key cryptography. While symmetric key encryption is done using 2-secret key, one key for encryption and another for decryption purpose. It is also termed as public key cryptography. The algorithm that uses asymmetric key encryption are RSA, PGP, SSH etc. Encryption requires two things viz an algorithm and a key. The strength of the encryption technique depends on its key [10]. The strong the key, then it will be more difficult to break the code.

2.3 Decryption Technique

Decryption is carried out to convert cipher text into original text. It is performed in the reverse manner of encryption [4]. Mainly it is performed at receiver's end so that it can extract the hidden data from an image.

3. PROPOSED TECHNIQUE

3.1 Zero Distortion Technique on Color Images

In the proposed technique matching is done between the common binary bits of text data and binary bits of pixels of cover image and storing the matched locations in a matrix format.

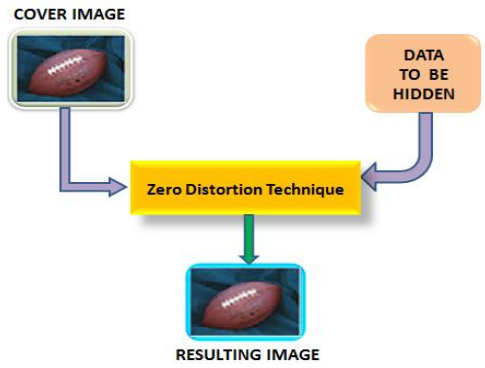

Figure.1. Embedding of text into image

This matrix is the location matrix and to provide more security this location matrix is encrypted using indexed based chaotic sequence which is very random and not easily breakable. Decryption will be carried out in reverse manner using the same procedure. This proposed technique provides us with zero distortion in the cover image and can hide data in all the three bands viz red, green and blue. So a large amount of data can be hidden with respect to gray images.

3.2 Encryption Technique

Chaos is a state of irregularity and disorder which makes the situation more complex to handle .Chaotic sequence has been used in the field of cryptography for providing randomness to the data so that it will become more secure. Chaotic sequences and the word random can be used interchangeably.

Properties of chaotic sequence which makes it more powerful are:

- Unpredictability
- In decomposability
- Element of regularity

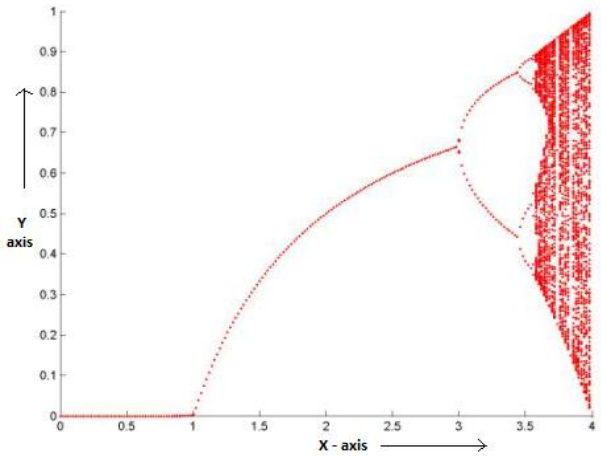

Figure.2. Bifurcation diagram for logistic map of μ

The system shows different behavior for different range of μ. Some of them are shown below.

- For $0 < \mu < 1$ the output of the system is zero irrespective of the initial condition.
- For $1 < \mu < 3$ the system reaches a steady state.
- For $3 < \mu < 1+\sqrt{6}$ (approximately $\mu = 3.45$) the system oscillates between two values.
- For $\mu > 1+\sqrt{6}$ the system oscillates between infinite values and shows chaotic behavior.
- Beyond $\mu = 4$, the values completely goes beyond the interval [0, 1] and shows divergence for almost all initial values.

Thus logistic map shown in fig .2 serves as an excellent chaotic system. A chaotic system is very sensitive to initial conditions. The logistic map between 3.57 to 4 shows chaotic behavior.

Chaotic sequence is so random that changing a single bit varies the sequence up to a large extent.

The value generated by the chaotic sequence varies in the time interval of [0, 1].

$$\text{Chaotic sequence} = \begin{cases} 0 \\ 1 \end{cases}$$

Calculation of the values of the sequence will be carried out by using the formula.

FORMULA:

$$X_{n+1} = \mu * X_n * (1-X_n)$$

Here the value of μ is constant. Initially we fix the value of X_0 in the range of [0, 1].

$$X_0 = \begin{cases} 0 \\ 1 \end{cases}$$

We had randomized the location matrix by using chaotic sequence.

3.3 Techniques for embedding and extracting text from image using Zero Distortion Technique on color images

We had extended our previous work by applying Zero Distortion Technique on color images [12].The idea is to hide more amount of data on color images in R G B planes as composed to gray images.

3.3.1 Algorithm for embedding text into color image

This technique will embed the input text into color image without altering any pixel value of the image.

INPUT: A color cover image (gray) **(I_0)**, an input text **(T_0).**

OUTPUT: Chaotic sequence **(C_{0s})** and stego-image **(I_S).**

1) Read the cover image **(I_0)** and the input text **(T_0).**

2) Convert the cover image **(I_0)** into RGB bands separately i.e. in decimal format and calculate the length of all bands.

3) Convert the RGB bands into binary format separately.

4) Convert the text data **(T_0)** to ASCII format and then into binary format.

5) Convert the matrix of text data **(T_0)** into a column vector.

6) Set the variable band=1

//Initially data will be hide in red band so variable band=1 means red band, 2 for green band and 3 means blue band

7) For all w=1 to 8 repeat steps 9 to 12

8) For i=1 to length_of_redband

//Firstly we will hide it in red band if data left we will proceed to another band i.e. green

9) Match the bits of text data (i.e. column vector) with the bits of the cover image **(I_0).**

10) If matched then save the location value in a matrix of n rows and 8 columns is the length of the text.

11) Sort the values generated by the above formula and extract the index of the sequence.

12) Increase count variable and location by 1.

13) Else increase the location.

// resulting is the matrix of location (L0S).Perform encryption on L_{0S} by using chaotic sequence approach.

14) If count ~=length_of_data then set band=2 and repeat step 9 to 12

15) If data left i.e. again count ~=length_of_data then set band=3 and repeat step 9 to 12

16) Calculate the dimension of matrix of location (L0S) m x n, where m is the rows and n is the column..

17) Set X0 value in the range of 0-1, and value of μ in the range of 3.5 – 4.

18) For i=1 to m x n

19) Apply the formula of chaotic sequence i.e.

20) Xn+1=μ * Xn * (1-Xn)

21) Sort the values generated by the above formula and extract the index of the sequence.

22) Reshape the indexes into m x n matrix same as the dimension of matrix of location (L_{0S}).

//This is the chaotic sequence i.e. C_{0S} which will be passed to the decryption side.

23) If count equals length of the text then text has been successfully embedded into image.

24) Else text has not been successfully embedded into image.

25) Display the stego-image (I_S).

3.3.2 Algorithm for extracting text from image

This algorithm will extract the hidden text (T_0) from the stego image (Is).

INPUT: Chaotic sequence (C_{0s}), value of X_0 and μ, stego image (Is).

OUTPUT: Embedded input text (T_0)

1) Extract the matrix of location (L_{0S}) from the chaotic sequence by using the same formula.

//Decryption has been performed

2) Convert the stego image (Is) pixels into RGB bands separately i.e. in decimal format and then in binary format respectively.

3) Calculate the length of RGB bands.

4) Firstly matching will be done with binary values of red band, if data left then proceed to green and blue band.

5) For i=1 to length of the red band repeat step from 3 to 5

6) Match the image red band location and value of the location matrix of stego image **(L_0s)**

7) If matched then save the value at that location of the image in an array.

8) Else the location of the image is increased.

9) If data left then repeat step 6 to 8 with green band

10) Else extract the bits from locations.

11) If data left then repeat step 6 to 8 with blue band

12) Else convert the array into decimal format i.e. the ASCII value of the embedded text.

13) Convert the ASCII value into characters and transpose it.

14) Embedded text will be displayed i.e. T_0.

3.4 Execution of Algorithm for embedding input data into cover image

It involves two steps:

1) Embedding text into color image and getting location matrix as the output.
2) Encrypt these locations by using chaotic sequence.

3.4.1 Embedding text into color image

We are taking peppers.png as the cover image (color) and the input text is 'Zero-Distortion'. We will proceed as per the algorithm.

STEP 1) Input the color cover image (I_0).

Figure. 3. Cover image peppers.png

STEP 2) Conversion of the image pixel (decimal format) into RGB bands separately (binary format). Image value in binary format has 49858 rows and 8 columns in each band.

red_band <26730x8 uint8>

	1	2	3	4	5	6	7	8
1	1	1	1	1	1	1	0	0
2	1	0	1	1	1	1	0	0
3	1	0	1	1	1	1	0	0
4	1	0	0	0	0	0	1	0
5	0	1	1	1	1	1	0	0
6	0	1	0	0	0	0	1	0
7	0	1	1	1	1	1	0	0
8	1	0	0	0	0	0	1	0
9	0	1	1	1	1	1	0	0
10	1	0	1	1	1	1	0	0
11	0	0	1	0	0	0	1	0
12	1	1	0	0	0	0	1	0
13	0	1	1	1	1	1	0	0
14	0	1	0	0	0	0	1	0
15	1	0	0	0	0	0	1	0
16	1	0	1	0	0	0	1	0

Figure.4. Cover image pixels in Red-Band (binary format)

green_band <26730x8 uint8>

	1	2	3	4	5	6	7	8
1	1	1	0	0	0	1	0	0
2	1	0	1	0	0	1	0	0
3	0	0	1	0	0	1	0	0
4	1	1	0	0	0	1	0	0
5	1	1	0	0	0	1	0	0
6	1	1	0	0	0	1	0	0
7	0	0	1	0	0	1	0	0
8	1	0	1	0	0	1	0	0
9	0	0	1	0	0	1	0	0
10	0	0	1	0	0	1	0	0
11	0	0	1	0	0	1	0	0
12	1	0	1	0	0	1	0	0
13	0	1	1	0	0	1	0	0
14	1	0	1	0	0	1	0	0
15	0	1	1	0	0	1	0	0
16	0	1	1	0	0	1	0	0

...

Figure.5. Cover image pixels in Green-Band (binary format)

blue_band <26730x8 uint8>

	1	2	3	4	5	6	7	8
1	0	0	0	0	0	0	1	0
2	0	1	0	0	0	0	1	0
3	1	0	1	1	1	1	0	0
4	1	1	1	1	1	1	0	0
5	0	0	0	0	0	0	1	0
6	1	0	0	0	0	0	1	0
7	0	0	0	0	0	0	1	0
8	0	0	1	0	0	0	1	0
9	0	1	0	0	0	0	1	0
10	1	1	1	1	1	1	0	0
11	0	1	0	0	0	0	1	0
12	0	1	0	0	0	0	1	0
13	1	1	1	1	1	1	0	0
14	1	0	1	1	1	1	0	0
15	0	0	1	0	0	0	1	0
16	1	0	1	0	0	0	1	0

...

Figure.6. Cover image pixels in Blue-Band (binary format)

STEP 3) Input the Text data **(T_0)**

'Zero-Distortion'

STEP 4) Conversion of text data into ASCII FORMAT

```
ASCII_VALUE_OF_TEXT =

    90   101   114   111    45    68   105   115   116   111   114   116   105   111   110
```

STEP 5) Conversion ASCII values into binary format. Text data in binary format has 15 rows and 8 columns.

Binary_Value_Of_Text <15x7 double>

	1	2	3	4	5	6	7
1	0	1	0	1	1	0	1
2	1	0	1	0	0	1	1
3	0	1	0	0	1	1	1
4	1	1	1	1	0	1	1
5	1	0	1	1	0	1	0
6	0	0	1	0	0	0	1
7	1	0	0	1	0	1	1
8	1	1	0	0	1	1	1
9	0	0	1	0	1	1	1
10	1	1	1	1	0	1	1
11	0	1	0	0	1	1	1
12	0	0	1	0	1	1	1
13	1	0	0	1	0	1	1
14	1	1	1	1	0	1	1
15	0	1	1	1	0	1	1

Figure.7. Binary format of text data

STEP 6) Matching of text bits with the bits of the red band. If matched save that location into matrix form (dimensions same as of text in binary format). If data remains left then hide this in green and blue band by using similar technique of matching. Location Matrix (L_{0S}) consists of 15 rows and 8 columns.

Matrix_Of_Locations <15x8 double>

	1	2	3	4	5	6	7	8
1	5	8	9	10	12	13	15	17
2	18	20	22	23	24	25	28	29
3	30	32	33	34	36	37	38	40
4	42	48	49	50	51	52	54	55
5	59	60	63	64	67	68	70	71
6	72	74	75	77	79	82	86	90
7	91	92	93	94	95	97	98	99
8	100	101	107	109	111	112	114	115
9	118	119	120	124	125	127	128	131
10	133	134	135	136	140	144	145	146
11	147	148	156	159	160	161	165	167
12	168	169	172	174	177	179	180	183
13	184	185	186	187	189	190	191	192
14	197	199	200	201	204	210	211	212
15	214	218	219	220	224	225	226	228

Figure.8. Location Matrix (L_{0S})

3.4.2 Encrypting the Location Matrix (L_{OS}) using Chaotic Sequence

STEP 7) We will use formula for calculating chaotic sequence:

$$X_{n+1}=\mu * X_n * (1-X_n)$$

Fix the value of X_0 in the range of [0, 1] i.e. for

- example 0.6 and the value of μ be 3.62.
- Calculate the dimension of the location matrix i.e. rows and column (m x n).
- Calculate the next (m x n) terms by using above formula.

Chaotic_Sequence <15x8 double>

	1	2	3	4	5	6	7	8
1	0.6000	0.8404	0.3286	0.8800	0.5331	0.8713	0.3316	0.7750
2	0.8700	0.4863	0.7972	0.3829	0.9023	0.4065	0.8035	0.6320
3	0.4100	0.9056	0.5860	0.8566	0.3196	0.8745	0.5724	0.8430
4	0.8769	0.3100	0.8794	0.4454	0.7883	0.3977	0.8872	0.4797
5	0.3914	0.7754	0.3843	0.8954	0.6049	0.8683	0.3627	0.9048
6	0.8635	0.6314	0.8577	0.3384	0.8664	0.4144	0.8379	0.3124
7	0.4274	0.8437	0.4423	0.8128	0.4197	0.8797	0.4924	0.7787
8	0.8871	0.4781	0.8942	0.5516	0.8829	0.3836	0.9060	0.6248
9	0.3630	0.9045	0.3430	0.8966	0.3748	0.8571	0.3086	0.8498
10	0.8382	0.3131	0.8169	0.3360	0.8495	0.4439	0.7735	0.4626
11	0.4916	0.7796	0.5422	0.8088	0.4635	0.8949	0.6352	0.9012
12	0.9060	0.6228	0.8998	0.5606	0.9014	0.3411	0.8400	0.3228
13	0.3087	0.8516	0.3269	0.8930	0.3221	0.8147	0.4872	0.7924
14	0.7736	0.4581	0.7976	0.3465	0.7915	0.5473	0.9057	0.5963
15	0.6348	0.8999	0.5852	0.8208	0.5982	0.8981	0.3097	0.8727

Figure.9. Random values generated by Chaotic Sequence

- Sort these values in increasing order.

Chaotic_Sequence_Sort <15x8 double>

	1	2	3	4	5	6	7	8
1	0.3086	0.3430	0.4423	0.5516	0.7735	0.8169	0.8664	0.8949
2	0.3087	0.3465	0.4439	0.5606	0.7736	0.8208	0.8683	0.8954
3	0.3097	0.3627	0.4454	0.5724	0.7750	0.8379	0.8700	0.8966
4	0.3100	0.3630	0.4581	0.5852	0.7754	0.8382	0.8713	0.8981
5	0.3124	0.3748	0.4526	0.5860	0.7787	0.8400	0.8727	0.8998
6	0.3131	0.3829	0.4635	0.5963	0.7796	0.8404	0.8745	0.8999
7	0.3196	0.3836	0.4781	0.5982	0.7883	0.8430	0.8769	0.9012
8	0.3221	0.3843	0.4797	0.6000	0.7915	0.8437	0.8794	0.9014
9	0.3228	0.3914	0.4863	0.6049	0.7924	0.8495	0.8797	0.9023
10	0.3266	0.3977	0.4872	0.6228	0.7972	0.8498	0.8800	0.9045
11	0.3269	0.4065	0.4916	0.6248	0.7976	0.8516	0.8829	0.9048
12	0.3316	0.4100	0.4924	0.6314	0.8035	0.8566	0.8871	0.9056
13	0.3360	0.4144	0.5331	0.6320	0.8088	0.8571	0.8872	0.9057
14	0.3384	0.4197	0.5422	0.6348	0.8128	0.8577	0.8930	0.9060
15	0.3411	0.4274	0.5473	0.6352	0.8147	0.8635	0.8942	0.9060

Figure.10. Sorted values generated by Chaotic Sequence

Extract the indexes of the above matrix and arrange them in mxn format.

Chaotic_Index <15x8 double>

	1	2	3	4	5	6	7	8
1	128	120	93	109	145	135	79	161
2	184	201	144	174	197	220	68	64
3	226	70	50	38	17	86	18	124
4	48	118	199	219	60	133	13	225
5	90	125	146	33	99	180	228	172
6	134	23	160	212	148	8	37	218
7	36	112	181	224	51	40	42	167
8	189	63	55	5	204	92	49	177
9	183	59	20	67	192	140	97	24
10	9	52	191	169	22	131	10	119
11	186	25	147	115	200	185	111	71
12	15	30	98	74	28	34	100	32
13	136	82	12	29	159	127	54	211
14	77	95	156	214	94	75	187	168
15	179	91	210	165	190	72	107	114

Figure.11. Indexes of the sorted values of Chaotic Sequence

This is the resultant matrix chaotic sequence matrix **(C_{0S})** which will be passed to the decryption side along with the stego image **(I_S).**

STEP 8) Stego image **(I_S)** has been displayed i.e. same as cover image **(I_0)** as we haven't alter any bit of the cover image, which leads to zero-distortion in the cover image and a large amount of data can be hidden in color cover.

Figure.12. Stego Image

3.5 Execution of Algorithm for extracting input data from stego image:

It involves two steps:

1) Chaotic sequence matrix **(C_{0S})** is decrypted into matrix of locations **(L_{0S}).**

2) Extraction of text data from stego image **(I_s).**

3.5.1 Chaotic sequence matrix (C_{0S}) is decrypted into matrix of locations (L_{0S})

STEP 1) Pass the stego image (IS) and the chaotic sequence matrix (C_{0S}) and the value of X_0 and μ i.e. between [0, 1].

Figure.13. Stego image

Chaotic_Index <15x8 double>

	1	2	3	4	5	6	7	8
1	128	120	93	109	145	135	79	161
2	184	201	144	174	197	220	68	64
3	226	70	50	38	17	86	18	124
4	48	118	199	219	60	133	13	225
5	90	125	146	33	99	180	228	172
6	134	23	160	212	148	8	37	218
7	36	112	181	224	51	40	42	167
8	189	63	55	5	204	92	49	177
9	183	59	20	67	192	140	97	24
10	9	52	191	169	22	131	10	119
11	186	25	147	115	200	185	111	71
12	15	30	98	74	28	34	100	32
13	136	82	12	29	159	127	54	211
14	77	95	156	214	94	75	187	168
15	179	91	210	165	190	72	107	114

Figure.14. Indexes of the sorted values of Chaotic Sequence

STEP 2) The value of X_0 in the range of [0, 1] same as encryption

side i.e. 0.6 and the value of μ be 3.62.

- Calculate the dimension of the location matrix i.e. rows and column (m x n).
- Calculate the next (m x n) terms by using the same formula.

$$X_{n+1=}\mu * X_n * (1-X_n)$$

- Compare the index of the sequence generated by the above formula with the chaotic sequence matrix **(C_{0s})** and arrange the accordingly in increasing order.
- The resultant is the location matrix **(L_{0S}).**

Matrix_Of_Locations <15x8 double>

	1	2	3	4	5	6	7	8
1	5	8	9	10	12	13	15	17
2	18	20	22	23	24	25	28	29
3	30	32	33	34	36	37	38	40
4	42	48	49	50	51	52	54	55
5	59	60	63	64	67	68	70	71
6	72	74	75	77	79	82	86	90
7	91	92	93	94	95	97	98	99
8	100	101	107	109	111	112	114	115
9	118	119	120	124	125	127	128	131
10	133	134	135	136	140	144	145	146
11	147	148	156	159	160	161	165	167
12	168	169	172	174	177	179	180	183
13	184	185	186	187	189	190	191	192
14	197	199	200	201	204	210	211	212
15	214	218	219	220	224	225	226	228

Figure15. Location Matrix (L_{0S})

3.5.2 Extraction of text data (T_0) from the stego image (I_s)

STEP 3) Conversion of the stego image **(Is)** pixel (i.e. in decimal) into binary format. Text data in binary format has 13 rows and 8 columns.

red_band <26730x8 uint8>

	1	2	3	4	5	6	7	8
1	1	1	1	1	1	1	0	0
2	1	0	1	1	1	1	0	0
3	1	0	1	1	1	1	0	0
4	1	0	0	0	0	0	1	0
5	0	1	1	1	1	1	0	0
6	0	1	0	0	0	0	1	0
7	0	1	1	1	1	1	0	0
8	1	0	0	0	0	0	1	0
9	0	1	1	1	1	1	0	0
10	1	0	1	1	1	1	0	0
11	0	0	1	0	0	0	1	0
12	1	1	0	0	0	0	1	0
13	0	1	1	1	1	1	0	0
14	0	1	0	0	0	0	1	0
15	1	0	0	0	0	0	1	0
16	1	0	1	0	0	0	1	0

Figure.16 . Stego image pixel of Red Band (binary format)

green_band <26730x8 uint8>

	1	2	3	4	5	6	7	8
1	1	1	0	0	0	1	0	0
2	1	0	1	0	0	1	0	0
3	0	0	1	0	0	1	0	0
4	1	1	0	0	0	1	0	0
5	1	1	0	0	0	1	0	0
6	1	1	0	0	0	1	0	0
7	0	0	1	0	0	1	0	0
8	1	0	1	0	0	1	0	0
9	0	0	1	0	0	1	0	0
10	0	0	1	0	0	1	0	0
11	0	0	1	0	0	1	0	0
12	1	0	1	0	0	1	0	0
13	0	1	1	0	0	1	0	0
14	1	0	1	0	0	1	0	0
15	0	1	1	0	0	1	0	0
16	0	1	1	0	0	1	0	0

Figure.17. Stego image pixel of Green Band (binary format)

blue_band <26730x8 uint8>

	1	2	3	4	5	6	7	8
1	0	0	0	0	0	0	1	0
2	0	1	0	0	0	0	1	0
3	1	0	1	1	1	1	0	0
4	1	1	1	1	1	1	0	0
5	0	0	0	0	0	0	1	0
6	1	0	0	0	0	0	1	0
7	0	0	0	0	0	0	1	0
8	0	0	1	0	0	0	1	0
9	0	1	0	0	0	0	1	0
10	1	1	1	1	1	1	0	0
11	0	1	0	0	0	0	1	0
12	0	1	0	0	0	0	1	0
13	1	1	1	1	1	1	0	0
14	1	0	1	1	1	1	0	0
15	0	0	1	0	0	0	1	0
16	1	0	1	0	0	0	1	0

Figure.18. Stego image pixel of Green Band (Binary Format)

- Text data in binary format has 15 rows and 8 columns

STEP 4) Match the location of value with the image red band locations and save the binary value at that location in a matrix having same dimension as of location matrix ($L_{0S.}$).

Binary_Value_Of_Text <15x7 double>

	1	2	3	4	5	6	7
1	0	1	0	1	1	0	1
2	1	0	1	0	0	1	1
3	0	1	0	0	1	1	1
4	1	1	1	1	0	1	1
5	1	0	1	1	0	1	0
6	0	0	1	0	0	0	1
7	1	0	0	1	0	1	1
8	1	1	0	0	1	1	1
9	0	0	1	0	1	1	1
10	1	1	1	1	0	1	1
11	0	1	0	0	1	1	1
12	0	0	1	0	1	1	1
13	1	0	0	1	0	1	1
14	1	1	1	1	0	1	1
15	0	1	1	1	0	1	1

Figure.19. Binary format of text data(T_0)

STEP 5) Conversion of the above matrix into decimal format i.e. ASCII value of the text. The ASCII value of the text data (T_0) is in column format.

ASCII_VALUE_OF_TEXT <15x1 double>

	1	2	3
1	90		
2	101		
3	114		
4	111		
5	45		
6	68		
7	105		
8	115		
9	116		
10	111		
11	114		
12	116		
13	105		
14	111		
15	110		

Figure.20. ASCII value of text data (T_0)

STEP 6) Transpose the column format (i.e. in row format).

```
ASCII_VALUE_OF_TEXT =

    90   101   114   111    45    68   105   115   116   111   114   116   105   111   110
```

Figure.21. Transpose of ASCII value of text data (T_0)

STEP 7) Conversion of ASCII format into character format and display the embedded text data (T_0).

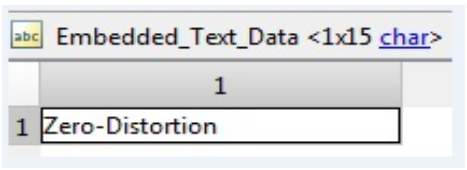

Figure.20.Text data T_0

FIGURES AND TABLES

4.1 Figures shows the schematic diagram of the Zero Distortion Technique on color image

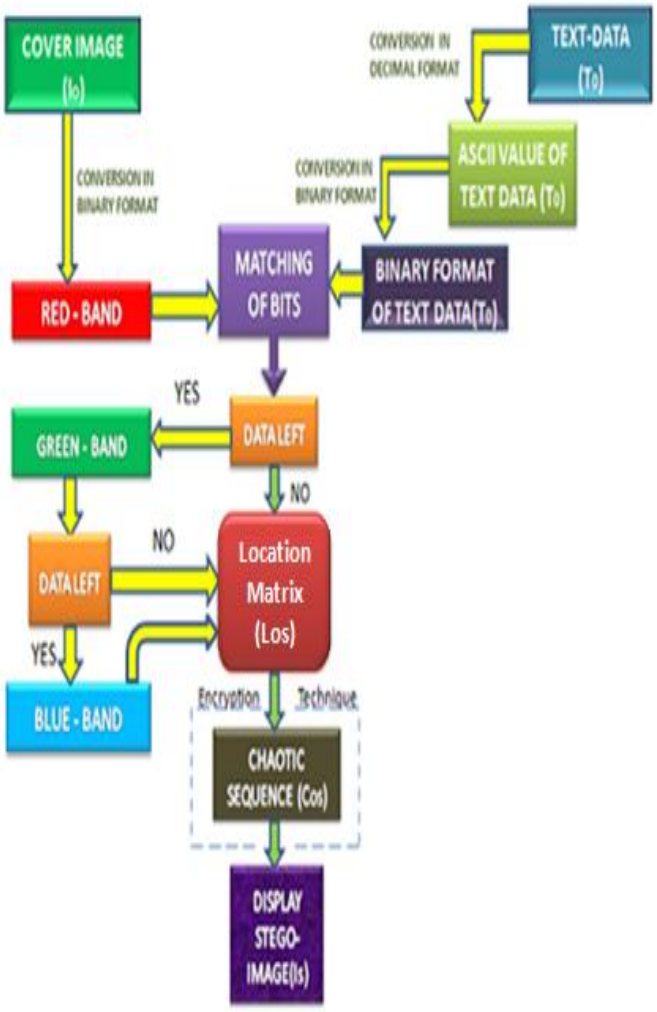

Figure.21. Schematic diagram for embedding text

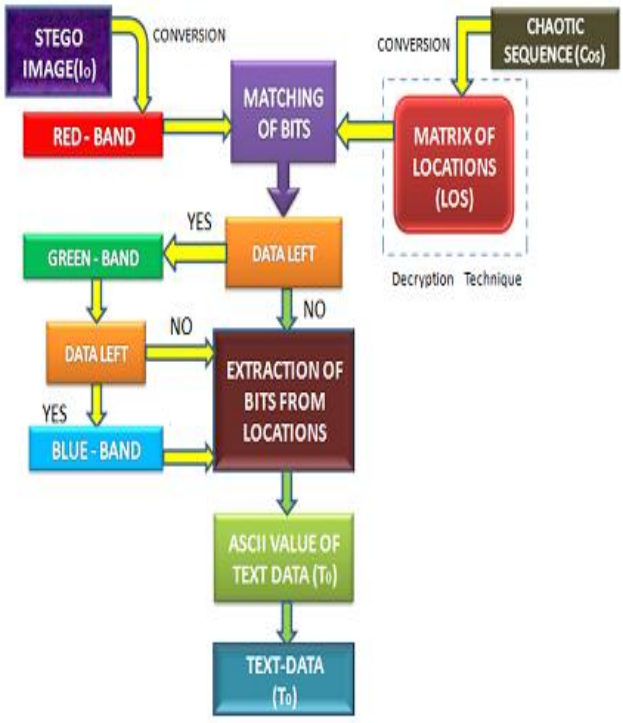

Figure.22. Schematic Diagram for extracting text

4.2 Comparison of LSB, Zero – Distortion Technique on gray images and color images

Comparison is based on the text length that we can hide in the image and time elapsed in hiding the text data. It is performed on various types of formats like jpg, png, tiff etc.

TABLE I. Implementation of LSB TECHNIQUE on all columns

IMAGE	ORIGINAL-SIZE	FORMAT	ROWS X COLUMNS	TEXT LENGTH	STEGOIMAGE SIZE	TIME(seconds)
Tire	41.7 KB	tif	205X232	5800	41.9 KB	9.632069
cameraman	52.6 KB	jpg	256X256	8192	52.8 KB	17.649408
rice	59.4 KB	png	256X256	8192	59.5 KB	18.743116
coins	51.5 KB	jpg	246X300	9000	52.5	20.074579
moon	114 KB	tiff	537X358	23986	117 KB	146.803198

TABLE II. Implementation of Zero distortion technique on Gray images on all columns

Image	Original-Size	Format	ROWS X COLUMNS	Image dimension	Text Length	StegoImage Size	Time(seconds)
Tire	41.7 KB	tif	205X232	<47560x8>	11856	41.7 KB	1.008306
cameraman	52.6 KB	jpg	256X256	<65536x8>	20338	52.6 KB	2.684668
rice	59.4 KB	png	256X256	<65536x8>	22905	59.4 KB	3.064316
coins	51.5 KB	jpg	246X300	<73800x8 uint8>	19552	51.5 KB	4.878285
moon	114 KB	tiff	537X358	<192246x8 uint8>	41922	114 KB	146.803198

Table III. Implementation of Zero distortion technique on color images on all columns

Image	Original-Size	Format	Rows x Columns	Image-Dimension	Text-Length	Stego-Image Size	Time(in seconds)
football	27 KB	jpg	256x320 x 3	81920 x 8	86049	27 KB	105.96
autumn	44 KB	tif	204x345 x 3	71070 x 8	67872	44 KB	65.72
onion	44 KB	png	135x198 x 3	26730 x 8	25387	44 KB	12
peppers	281 KB	png	194x320 x 3	49859 x 8	34418	281 KB	16.68

5. CONCLUSION

In this paper we had suggested a technique that is very powerful in comparison to other existing techniques. We can hide large amount of data and time complexity is less. This technique is compatible with all file formats. Indexed based chaotic sequence is used for encryption technique which provides more security to our data. Proposed technique is robust and providing zero distortion such that the stego image is similar to cover image.

6. ACKNOWLEDGMENT

We would also like to thank our friends, family members for their extreme support and blessings.

7. REFERENCES

[1] Moerland, T.“Steganography and Steganalysis”.Leiden Institute of Advanced Computing Science, www.liacs.nl/home/ tmoerl/privtech.pdf

[2] Wang, H & Wang, S.“Cyber warfare: Steganography vs. Steganalysis”, Communications of the ACM,47:10, October 2004

[3] W. Bender, N. Morimoto, A. Lu.”Techniques for data hiding”.IBM Syst. J. 35 (3/4) (1996) 313–336.

[4] G. Viji and J. Balamurugan.”LSB Steganography in Color and Grayscale Images without using the Transformation”.Bonfring International Journal of Advances in Image Processing, Vol. 1, Special Issue, December 2011

[5] R.J. Andersen and F.A.P. Petitcolas.“On the Limits of Steganography,” IEEE J. Selected Areas in Comm., vol.16, no. 4, 1998, pp. 474-481.

[6] Fabien A. P. Petitcolas, Ross J. Anderson and Markus G. Kuhn. “Information Hiding – A Survey”.Proceedings of the IEEE, special issue on protection of multimedia content, pp. 1062-1078, July 1999.

[7] Morkel.T, J.H.P. Eloff, M.S. Olivier.”An Overview of Image Steganography”.Proceedings of the Fifth Annual Information Security South Africa Conference , Sandton, South Africa 2005.

[8] William Stalling, Cryptography and network security: Principles and Practices (4th edition), Prentice 2006, ISBN – 978-81-775-8774-6.

[9] "Advanced Encryption Standard "(AES), National Institute of Standards and Technology (NIST), U.S. FIPS PUB 197 (FIPS 197),2001

[10] S. Batham, A. Acharya, V.K. Yadav, R.Paul.“A New Video Encryption Algorithm Based on Indexed Based Chaotic Sequence”.CONFLUENCE-2013, IET digital library.

[11] R. Paul, A. Acharya, V.K.Yadav, S. Batham.“Hiding Large Amount of Data using a New Approach of Video Steganography”.CONFLUENCE-2013, IET digital library.

Weburl:http://digitallibrary.theiet.org/content/conferences/10.1049/cp.2013.2338

[12] Shivani, Yadav. V. K., Batham.S, “An Approach to Image Steganography using Strength of Indexed Based Chaotic Sequence”, SSCC-2014 (Springer).

[13] Saumya Batham, V. Yadav, Amit Kumar Mallik. “ICSECV: An Efficient Approach of Video Encryption”. Published in: Contemporary Computing (IC3), 2014 Seventh International Conference. Date of Conference: 7-9 Aug. 2014, Pages: 425 - 430, Publisher: IEEE.

[14] V.K.Yadav, Saumya Batham, Anuja Acharya, Rahul Paul. “Approach to Accurate Circle Detection: Circular Hough Transform and Local Maxima Concept”. Electronics and Communication Systems (ICECS), 2014 International Conference, Publication Year: 2014, Page(s): 1 - 5, IEEE.

[15] V. K. Yadav, Saumya Batham, Amit Kumar Mallik. “False Circle Detection Algorithm based on Minimum Support Percentage and Euclidean Distance”. Emerging Trends and Applications in Computer Science (ICETACS), 2013 International Conference, Publication Year: 2013 , Page(s): 70 - 73, IEEE