ISSN: 2278 – 1323
International Journal of Advanced Research in Computer Engineering & Technology
Volume 2, Issue 1, January 2013

Some New Steganographic Techniques using Spatial Resolution Reduction

H. Faheem Ahmed and U. Rizwan

***Abstract*— In this paper, we present four new techniques to hide a large volume of text in a gray level image by reducing its spatial resolution to its half, to its quarter and by reducing the spatial resolution in width and in height. The actual image and the data embedded stego-images using the techniques are given. The Mean Square Error (MSE) and Peak to Signal Noise Ratio (PSNR) values have been determined. Histograms for the computed values of MSE and PSNR indices and embedding capacity are drawn.**

***Index Terms* — Steganography, Spatial Resolution Reduction, Mean Square Error, Peak to Signal Noise Ratio .**

I. INTRODUCTION

Sampling is the principal factor determining the spatial resolution of an image. Basically spatial resolution is the smallest discernible detail in an image. The measure of how closely lines can be resolved in an image is called *spatial resolution*, and it depends on properties of the system creating the image, not just the pixel resolution in pixels per inch (ppi). For practical purposes the clarity of the image is decided by its spatial resolution, not the number of pixels in an image. In effect, spatial resolution refers to the number of independent pixel values per unit length. Spatial resolution is a term that refers to the number of pixels utilized in construction of a digital image. Images having higher spatial resolution are composed with a greater number of pixels than those of lower spatial resolution. In short, spatial resolution is the density of pixels over the image: the greater the spatial resolution, the more pixels are used to display the image. We can experiment with spatial resolution to hide very huge amount of data.

II. REDUCING THE (SPATIAL RESOLUTION) IMAGE TO ITS HALF

Consider a 256 x 256 gray scale image. As the neighboring pixels are similar values in uniform region except boundaries, we can reduce the image to half of its size and use pixel values other than the rectangle ones to hide the text data. By taking out every other row and every other column, thus leaving only those matrix elements whose row and column indices are even:

$$\begin{array}{ccccccc} x_{11} & x_{12} & x_{13} & x_{14} & x_{15} & x_{16} & \cdots \\ x_{21} & \boxed{x_{22}} & x_{23} & \boxed{x_{24}} & x_{25} & \boxed{x_{26}} & \cdots \\ x_{31} & x_{32} & x_{33} & x_{34} & x_{35} & x_{36} & \cdots \\ x_{41} & \boxed{x_{42}} & x_{43} & \boxed{x_{44}} & x_{45} & \boxed{x_{46}} & \cdots \\ x_{51} & x_{52} & x_{53} & x_{54} & x_{55} & x_{56} & \cdots \\ x_{61} & \boxed{x_{62}} & x_{63} & \boxed{x_{64}} & x_{65} & \boxed{x_{66}} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{array}$$

By this technique, about 75% of pixels values can be changed, giving a hiding capacity of 75 percent and retaining only 25 percent of image as shown below.

$$\begin{array}{cccc} x_{22} & x_{24} & x_{26} & \cdots \\ x_{42} & x_{44} & x_{46} & \cdots \\ x_{62} & x_{64} & x_{66} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{array}$$

Consider the following 16x16 image, which we refer as Matrix -1.

1	17	33	49	65	81	97	113	129	145	161	177	193	209	225	241
2	18	34	50	66	82	98	114	130	146	162	178	194	210	226	242
3	19	35	51	67	83	99	115	131	147	163	179	195	211	227	243
4	20	36	52	68	84	100	116	132	148	164	180	196	212	228	244
5	21	37	53	69	85	101	117	133	149	165	181	197	213	229	245
6	22	38	54	70	86	102	118	134	150	166	182	198	214	230	246
7	23	39	55	71	87	103	119	135	151	167	183	199	215	231	247
8	24	40	56	72	88	104	120	136	152	168	184	200	216	232	248
9	25	41	57	73	89	105	121	137	153	169	185	201	217	233	249
10	26	42	58	74	90	106	122	138	154	170	186	202	218	234	250
11	27	43	59	75	91	107	123	139	155	171	187	203	219	235	251
12	28	44	60	76	92	108	124	140	156	172	188	204	220	236	252
13	29	45	61	77	93	109	125	141	157	173	189	205	221	237	253
14	30	46	62	78	94	110	126	142	158	174	190	206	222	238	254
15	31	47	63	79	95	111	127	143	159	175	191	207	223	239	255
16	32	48	64	80	96	112	128	144	160	176	192	208	224	240	256

Manuscript received on Jan 7, 2013.

H. Faheem Ahmed, Department of Computer Science, Islamiah College, Vaniyambadi 635 752, India.

U. Rizwan, Department of Mathematics, Islamiah College, Vaniyambadi 635 752, India

ISSN: 2278 – 1323
International Journal of Advanced Research in Computer Engineering & Technology
Volume 2, Issue 1, January 2013

Extracting the elements of rows and columns having even suffices, we arrive at the matrix of half of its size in both width and height (8x8).

18	50	82	114	146	178	210	242
20	52	84	116	148	180	212	244
22	54	86	118	150	182	214	246
24	56	88	120	152	184	216	248
26	58	90	122	154	186	218	250
28	60	92	124	156	188	220	252
30	62	94	126	158	190	222	254
32	64	96	128	160	192	224	256

Excluding the above pixel locations, the remaining pixels marked with X are used for embedding text data.

X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
X	18	X	50	X	82	X	114	X	146	X	178	X	210	X	242
X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
X	20	X	52	X	84	X	116	X	148	X	180	X	212	X	244
X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
X	22	X	54	X	86	X	118	X	150	X	182	X	214	X	246
X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
X	24	X	56	X	88	X	120	X	152	X	184	X	216	X	248
X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
X	26	X	58	X	90	X	122	X	154	X	186	X	218	X	250
X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
X	28	X	60	X	92	X	124	X	156	X	188	X	220	X	252
X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
X	30	X	62	X	94	X	126	X	158	X	190	X	222	X	254
X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
X	32	X	64	X	96	X	128	X	160	X	192	X	224	X	256

Note here that out of 256 (= 16x16) pixel values, only select 64 (= 8x8) are retained and the remaining 192 (= 256 – 64) pixels can be utilized for hiding the text.

For a 256 x 256 8-bit grey scale image, the effective resolution of the new image is only 128 x 128. That is, out of 65536 pixels required for storing the original image, only 16384 pixels are retained, and the remaining 49152 pixel positions are utilized for hiding the data, there by achieving 75 percent of embedding capacity. The algorithm to carry out this procedure is given below.

Algorithm

```
1. Read the image
2. Let r,c be the size of the image
   (width,height)
      for i=1:rows
          for j=1:cols
              if( i is odd or j is odd )
                   Embed a character in
     image pixel location i,j
              end
             end
      end
3. Display the stego image (of alternate
   rows and columns)
4. end
```

The original image (cameraman.tif) Figure 1, the stego image embedded with 49152 characters of text (Figure 2) and the stego image reduced to a size half its original dimensions (Figure 3) and the stego image in figure 3 doubled in dimensions (Figure 4) are given below.

Fig 1. Original image 256x256

Fig 2. Stego image embedded with 49152 characters

Fig 3. Reduced stego image to its ½

ISSN: 2278 – 1323
International Journal of Advanced Research in Computer Engineering & Technology
Volume 2, Issue 1, January 2013

Fig 4. Enlarged Stego image

III. Reducing the (Spatial Resolution) Image to its Quarter

Consider a 256 x 256 gray scale image. As the neighboring pixels are similar values in uniform region except boundaries, we can reduce the image to one-quarter of its size and use pixel values other than the boxed ones to hide the text data. By taking out every other row and every other column, thus leaving only those matrix elements whose row and column indices are even:

x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}	x_{17}	x_{18}	x_{19}	$x_{1,10}$	$x_{1,11}$	$x_{1,12}$	...
x_{21}	x_{22}	x_{23}	x_{24}	x_{25}	x_{26}	x_{27}	x_{28}	x_{29}	$x_{2,10}$	$x_{2,11}$	$x_{2,12}$	...
x_{31}	x_{32}	x_{33}	x_{34}	x_{35}	x_{36}	x_{37}	x_{38}	x_{39}	$x_{3,10}$	$x_{3,11}$	$x_{3,12}$	...
x_{41}	x_{42}	x_{43}	x_{44}	x_{45}	x_{46}	x_{47}	x_{48}	x_{49}	$x_{4,10}$	$x_{4,11}$	$x_{4,12}$	...
x_{51}	x_{52}	x_{53}	x_{54}	x_{55}	x_{56}	x_{57}	x_{58}	x_{59}	$x_{5,10}$	$x_{5,11}$	$x_{5,12}$	...
x_{61}	x_{62}	x_{63}	x_{64}	x_{65}	x_{66}	x_{67}	x_{68}	x_{69}	$x_{6,10}$	$x_{6,11}$	$x_{6,12}$	...
x_{71}	x_{72}	x_{73}	x_{74}	x_{75}	x_{76}	x_{77}	x_{78}	x_{79}	$x_{7,10}$	$x_{7,11}$	$x_{7,12}$	...
x_{81}	x_{82}	x_{83}	x_{84}	x_{85}	x_{86}	x_{87}	x_{88}	x_{89}	$x_{8,10}$	$x_{8,11}$	$x_{8,12}$	...
x_{91}	x_{92}	x_{93}	x_{94}	x_{95}	x_{96}	x_{97}	x_{98}	x_{99}	$x_{9,10}$	$x_{9,11}$	$x_{8,12}$	...
$x_{10,1}$	$x_{10,2}$	$x_{10,3}$	$x_{10,4}$	$x_{10,5}$	$x_{10,6}$	$x_{10,7}$	$x_{10,8}$	$x_{10,9}$	$x_{10,10}$	$x_{10,11}$	$x_{8,12}$	...
$x_{11,1}$	$x_{11,2}$	$x_{11,3}$	$x_{11,4}$	$x_{11,5}$	$x_{11,6}$	$x_{11,7}$	$x_{11,8}$	$x_{11,9}$	$x_{11,10}$	$x_{11,11}$	$x_{8,12}$	...
$x_{12,1}$	$x_{12,2}$	$x_{12,3}$	$x_{12,4}$	$x_{12,5}$	$x_{12,6}$	$x_{12,7}$	$x_{12,8}$	$x_{12,9}$	$x_{12,10}$	$x_{12,11}$	$x_{8,12}$	
...	...	...	...	...	...	...	...	...	...	...	...	...

By this technique, about 93.75 percent pixels values can be changed, giving a hiding capacity of 93.75 percent and retaining only 6.25 percent of image as shown below.

x_{33}	x_{37}	$x_{3,11}$	$x_{11,15}$
x_{73}	x_{77}	$x_{7,11}$	$x_{11,15}$
$x_{11,3}$	$x_{11,7}$	$x_{11,11}$	$x_{11,15}$
$x_{15,3}$	$x_{15,7}$	$x_{15,11}$	$x_{15,15}$

From Matrix -1 of section 2, retaining every fourth row and fourth column values from 3rd row 3rd column, we get following 4 x 4 matrix comprising of 16 pixel values and the remaining 240 (= 256 – 16) values are used for embedding.

35	99	163	227
39	103	167	231
43	107	171	235
47	111	175	239

Excluding the above pixel locations, the remaining pixels marked with X are used for embedding text data.

X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
X	X	35	X	X	X	99	X	X	X	163	X	X	X	227	X
X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
X	X	39	X	X	X	103	X	X	X	167	X	X	X	231	X
X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
X	X	43	X	X	X	107	X	X	X	171	X	X	X	235	X
X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
X	X	47	X	X	X	111	X	X	X	175	X	X	X	239	X
X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

For a 256 x 256 8-bit grey scale image, by reducing to one-quarter, the effective resolution of this new image is only 64 x 64. That is, out of 65536 pixels required for storing the original image, only 4096 pixels are retained, and the remaining 61440 pixel positions are utilized for data hiding there by achieving 93.75 percent of embedding capacity.

The stego image embedded with 61440 characters of text (Figure 5) and the stego image reduced to a size ¼ its original dimensions (Figure 6) and the stego image in figure 6 enlarged to 4 times in dimensions (Figure 7) are given below.

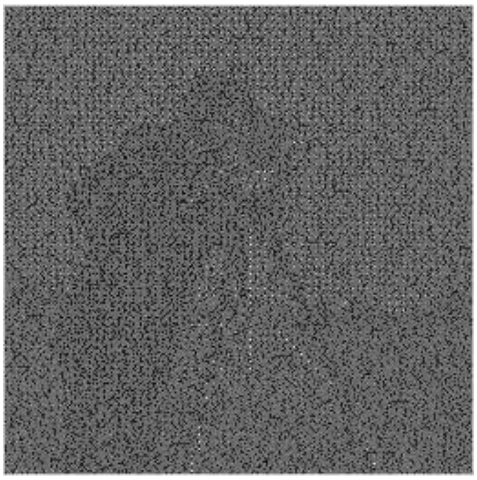

Fig 5. Stego image embedded with 61440 characters

Fig 6. Reduced stego image to its quarter

Fig 7. Enlarged Stego image

IV. Reducing the Spatial Resolution of Width (or Height) to its Half

By embedding data in alternate rows (or columns) and displaying image pixels in unaltered rows (or columns), the image is reduced to half in width (or in height) as shown below. By this technique 50 percent of pixel values can be changed. In the above example of 256 x 256, by this method, a total of 32768 characters are hidden and at the same time the quality of image is maintained except the width (or height).

The figures Fig 8 to Fig 9 give a view of the data embedded actual sized stego image and reduced stego image reduced to its width in one-half in size.

The figures Fig 10 to Fig 11 give a view of the data embedded actual sized stego image and reduced stego image reduced to its height in one-half in size.

Fig 8. Stego image embedded with 32768 characters

Fig 9. Reduced stego image to its Half in width

Fig 10. Stego image embedded with 32768 characters

Fig 11. Reduced stego image to its Half in Height

The Peak Signal to Noise Ratio (PSNR), Mean Square Error (MSE) are performance parameters to measure the quality of image.

- MSE: It is defined as square of error between cover stego-image. The error indicates the distortion in an image.MSE can be calculated by using two dimensional mathematical equation described as follows:

$$MSE = \left(\frac{1}{N}\right)^2 \sum_{i=1}^{M}\sum_{j=1}^{N}(X_{ij} - \bar{X}_{ij})^2$$

where Xij = the value of pixel in cover image and $\bar{X}_{ij}$ =the value of pixel in stego-image and N is the size of image.

- PSNR: It is a measure of quality of image. PSNR can be calculated by using the mathematical formula given below:

$$PSNR = 10 \, x \log \frac{255^2}{MSE} \, db$$

Table 1. The computed values of PSNR and MSE

No.	Spatial Resolution Reduction	PSNR	MSE	No. of bytes hidden
1.	To one-half	33.3266	30.2289	49152
2.	To one- quarter	30.0236	64.6732	61440
3.	To one-half in width	32.0679	40.3913	32768
4.	To one-half in Height	32.0594	40.4708	32768

The histograms of the computed values of PSNR and MSE indices for various techniques developed in this paper are presented in Fig.12. It is observed that the PSNR value is least for one-quarter spatial resolution reduction technique, whereas the corresponding MSE value is highest for this technique. Further, the amount of data hidden, using this technique, is high when compared with the other techniques.

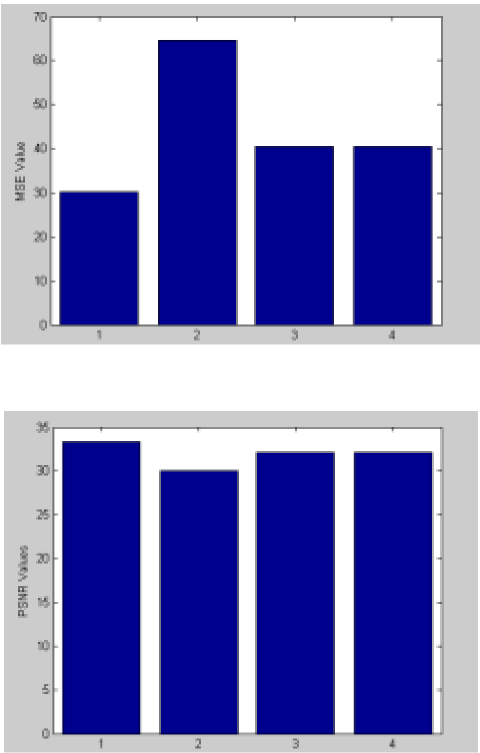

Fig .12 Histogram of the computed MSE and PSNR Values

The histogram representing the embedding capacity, while using each of the four techniques developed in this paper is presented in Figure 13. It is observed here that the spatial resolution reduction embedding technique to its one-quarter yield the optimum result.

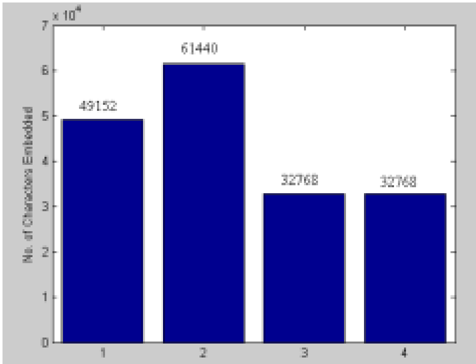

Fig 13. Embedding capacity

V. Conclusion

In this paper, we have proposed four new techniques for embedding text in a gray level image by reducing the spatial resolution of the image. Several examples have been included to explain the concept in detail. An actual image 256 x 256 `cameraman.tif` was taken for implementing the above methods and document of 30 pages was embedded and retrieved. The actual image along with data embedded stego image are shown explicitly for each method.

REFERENCES

[1] Cachin. C., An information-theoretic model for steganography, *Information and Computation*, Vol. **192** (1), Ed. Academic, USA, pp. 41 – 56, 2004.

[2] Faheem Ahmed, H and Rizwan. U, *An Alternative Technique in Data Embedding*, Advanced Materials in Physics, pp 233 – 242, 2012

[3] Fridrich, J and M. Goljan and D. Hogea and D. Soukal, Quantitative steganalysis of digital images: estimating the secret message length, *Multimedia Systems Journal* - Special issue on Multimedia Security, Vol. **9** (3), pp. 288 – 302, 2003.

[4] Rafael Gonzalez and Richard E. Woods, *Digital Image Processing*, Addison-Wesley, second edition, 2002.

[5] Rizwan. U and Faheem Ahmed. H, Comprehensive study on various types of steganographic schemes and possible steganalysis methods for various cover carrier like image, text, audio and video, *International Journal of Scientific and Engineering Research*, Volume **3,** Issue 11, November 2012, pp 151 – 154.

[6] Rizwan. U and Faheem Ahmed. H, A New Approach in Steganography using different Algorithms and Applying Randomization Concept, *International Journal of Advanced Research in Computer and Communication Engineering*, Vol.**1** (9), November 2012, pp 233 – 242.

Authors Biography

H. Faheem Ahmed earned his M.Tech. degree in Information Technology from Punjabi University and M.Phil. degree in Computer Science from Manonmaniam Sundaranar University. He is pursuing Ph.D. in Computer Science. He has guided 50 M. Phil. Research scholars in Computer Science. He is currently serving as the Associate Professor and Head of the Department of Computer Science and Applications, Islamiah College, Vaniyambadi and is serving the institution for the past 28 years. His research interest includes Steganography and Image processing. He has published 8 research articles and authored one book.

U. Rizwan earned his Ph.D. degree in Mathematics from the University of Madras. He is currently the Head of the Department of Mathematics, Islamiah College, Vaniyambadi and is serving the Institution for the past 26 years. He has published 43 research articles in journals of international repute. He has authored 7 books and is also the editor of two international journals. He has guided 30 M.Phil. Mathematics research scholars and one M. Tech. (IT) candidate. He is guiding Ph. D. research scholars in Mathematics and Computer Science. His research interest includes Image Processing, Hacking algorithms, Stochastic Processes, Fuzzy Logic, etc. He is the member of the board of studies in PG Mathematics of Thiruvalluvar University, Vellore. He is also an academic auditor.