International Journal of Soft Computing 13 (2): 38-47, 2018
ISSN: 1816-9503

Reversible Data Hiding of Images using Trinary Representation

[1]A. Rasmi and [2]M. Mohanapriya
[1]Karpagam Academy of Higher Education, Karpagam University, Tamil Nadu, India
[2]Department of Computer Science and Engineering and IT,
Coimbatore Institute of Technology, Coimbatore, Tamil Nadu, India

Abstract: Reversible data hiding with high capacity is a challenging area of digital image steganography. Prediction error histogram of images have large frequency of zero values. The modification of prediction error histograms can be utilized to store secret digits. This study presents a method called to encode a secret image into digits in base and these are called trits which hides it by modifying the prediction error histogram of the cover image. This improves the embedding capacity while allowing the original cover image to be recovered with zero errors. Experimental results indicate that the secret image can be recovered with little loss. The peak signal to noise ratio of the recovered secret image is above 30 decibels. The embedding capacity is high enough to allow the hiding of 80×80 secret grayscale images inside 256×256 cover grayscale images. The method is applicable to color images also. Several predictors are considered in this research and their relative performance are analyzed and discussed in detail. The performance of the method is compared against several existing reversible hiding techniques. The method has low computational complexity with fast execution times.

Key words: Steganography, reversible data hiding, grayscale, embedding capacity, least significant bit, histogram, cover image, peak signal to noise ratio

INTRODUCTION

Steganography is the technique of hiding a secret signal inside a cover signal such that the distortion of the cover signal is not visible (Provos and Honeyman, 2003). The requirement of steganography is more stringent than cryptography. The hidden data must not be accessible to unintended observers as in the case of cryptography. In addition, the presence of hidden data must not be revealed to the observers. Digital images are ideal cover signals due to the large redundancy available in image data and also due to the presence of large number of digital images in public networks. The reversibility or invertibility condition implies that the original cover image must be recovered by the receiver without any data loss (Lee *et al*., 2016). Reversible data hiding has many applications in the field of medical images and remote sensing images where precision of cover image is important. Several reversible data hiding techniques are available in literature. An important drawback of existing systems are the low embedding capacity available.

There are three main objectives in reversible data hiding. First the cover image must be recovered by the receiver without any errors. Secondly, the visual distortion of the cover image due to the embedding of data must be minimal. Thirdly the presence of hidden data must not be detected by statistical analysis of the data. In this research, the hiding of secret images within cover images is addressed. This requires a large embedding capacity with the usual low distortion and reversibility criteria.

Literature review: The most common data hiding in images are the Least Significant Bit (LSB) based steganographic systems (Sarreshtedari and Akhaee, 2013). One or more LSBs are overwritten with the secret bits. The original cover image signal is irreversibly lost in all these cases. In a spread spectrum based methods (Gkizeli *et at*., 2007) due to rounding off errors the original cover image is not recoverable. In Quantization Index Modulation (QIM) methods (Kalantari and Ahadi, 2010; Guillemot and Moureaux, 2010), the quantization errors are introduced preventing the recovery of cover image. Usually a compression technique has to be combined with steganography to achieve reversible embedding.

The simplest method of reversible data hiding in images is the modulo -256 addition of image hash to the pixel values for the purpose of authentication (Honsinger *et al*., 2001). Another authentication technique uses lossless multiresolution transform and patchwork (Bender *et al*., 1996). The lossless compression of bitplanes of cover image is used to provide room for secret data and overhead book keeping data (Fridrich *et al*., 2001). Authentication methods require small embedding capacity and are not suitable for hiding secret images. Fridrich *et al*. (2001) introduced a reversible

Corresponding Author: A. Rasmi, Karpagam Academy of Higher Education, Karpagam University, Tamil Nadu, India

embedding technique by Goljan *et al.* (2001). The image blocks are classified into three classes, namely regular R, singular S and unusable U. The R and S blocks can be converted into one another and are used as carrier of secret bits. The secret bits are compressed and embedded in the R, S bit sequence of the cover image. This method improved the embedding capacity but the distortion caused by the R-S inter conversion may be too visible. Integer Wavelet Transforms (IWT) are utilized by Xuan *et al.* (2002) to embed large amounts of data. The rounding off errors in traditional wavelet transforms are avoided in IWT and the image signal is highly decorrelated. The IWT coefficients in high frequency subbands are compressed to create room for secret bits. The IWT method increased the embedding capacity to 0.36 bits per pixel (bpp) from 0.16 bpp by Goljan *et al.* (2001). Quantization is used by Celik *et al.* (2005) and the residuals are highly compressed using CALIC lossless image compression algorithm to allow embedded data of up to 0.56 bpp. Ni *et al.* (2006) utilized histogram modification to allow up to 0.31 bpp while keeping the distortion very low. A high PSNR value of 48 dB is achieved whereas the PSNR values in the other methods are below 38 dB. Chen and Chang (2010) utilized Side Match Vector Quantization (SMVQ) with a codebook to achieve reversible embedding. Hsu *et al.* (2014) introduced a new application of reversible and visible watermarks for ownership identification. Wu and Lin (2003) introduced multilevel reversible data hiding scheme based on difference image histogram modification. Their observation was that the difference image had large number of pixels falling in the peak value. It used the peak points to hide messages and achieved more than 1 bpp with PSNR above 35 dB. Fallahpour and Sedaaghi (2007) reported 0.55 bpp with PSNR above 40 dB using the Shifted Gradient Adjusted Prediction Error (SGAPE) histograms. Tai *et al.* (2009) used distribution of pixel differences to achieve large hiding capacity of up to 0.17 bpp with PSNR of more than 48 dB. Tsai *et al.* (2009) used the prediction error histogram to hide secret bits by shifting the values between the peak and zero histogram points. This allowed an improvement of 1.5 dB in the PSNR values. Zhang (2012) introduced separable reversible data hiding in encrypted images by dividing the encrypted image into blocks and computing block smoothness. This research was further enhanced by Hong *et al.* (2012) with a better side atch scheme that took the correlations of pixel values in neighboring blocks into account.

This study is based on the work by Lee *et al.* (2005). The contribution of this study is the increased embedding capacity suitable for hiding a secret image within a cover image. The secret image pixels are encoded in trinary representation with 5 trits per pixel. Then the peak value of prediction error histogram and the two adjoining histogram positions are used to represent trits.

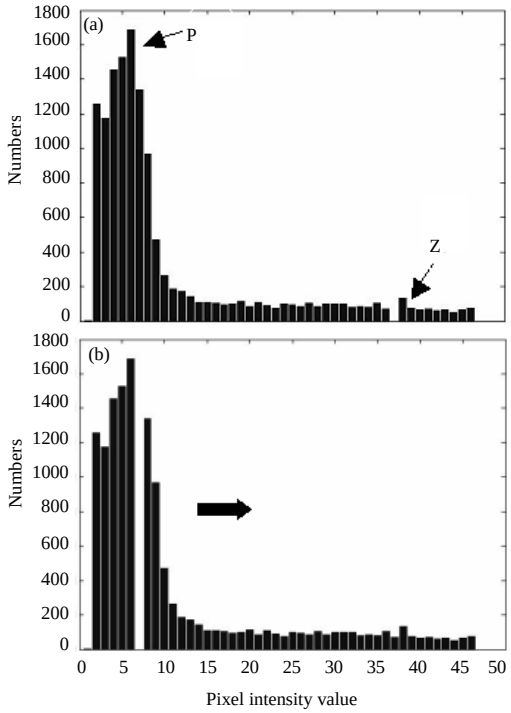


Fig. 1: Illustration of histogram modification: a) Original histogram and b) Modified histogram

Existing system: The histogram modification method described by Ni *et al.* (2006) starts either with the image histogram or prediction error histogram. Let, H be the histogram used. H(v) is a mapping from the value v to the number of occurrences of value v. In case of prediction error, v ranges from -255 to +255. Let the maximum value of H occur at the value of v = p and a suitable zero value occur at v = z. The members of histogram bars from next to p up to before z are shifted towards z. This process creates an empty bin next to p. The values at p are perturbed to the empty location depending on the message bits (Fig. 1).

More specifically, let the message binary sequence be M. Then, the pixels falling at the value p are perturbed by the following equation:

$$p \rightarrow \begin{cases} p \text{ if } M = 0 \\ p+1 \text{ if } M = 1 \end{cases} \quad (1)$$

In case the zero bin falls to the left of the peak bin, then the pixel is decremented instead of incrementing. The capacity of embedding is equal to the peak value of the histogram:

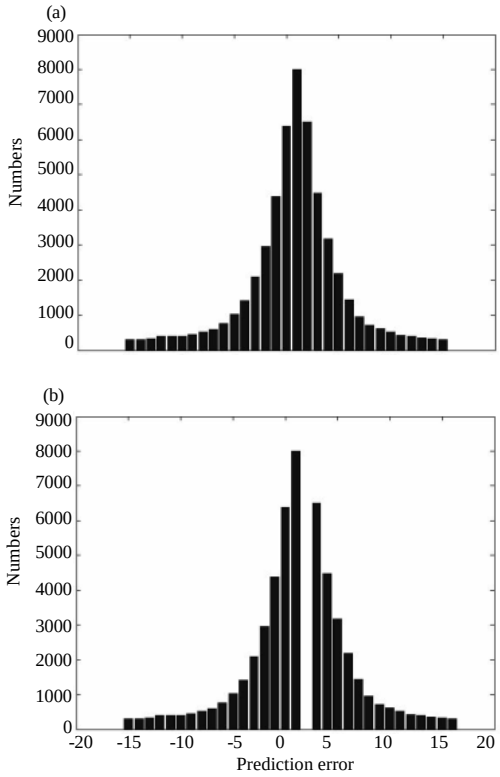


Fig. 2: Prediction error histogram modification illustration; a) Original historam and b) Mondified histogram

$$\text{Capacity} = H(p) \text{ bits} \quad (2)$$

The process can be reversed by remembering the peak value. In prediction error histogram the peak value occurs close to zero as shown in Fig. 2. Then the modification can take place by either shifting the bins to the left or right.

The advantages of using prediction error histogram is the high peak value at zero that is usually obtained if the predictor is good. There are no overheads to be remembered. Also the bins can be shifted to the left or right all the way out of the range without causing any overflows/underflows.

MATERIALS AND MATHODS

Pixel prediction is used widely in lossless as well as lossy image compression algorithms to take advantage of the high correlations in pixel intensity values.

The pixel value marked x in Fig. 3 can be predicted using its previous neighbors in top to bottom left to right scan order using JPEG-LS nonlinear predictor:

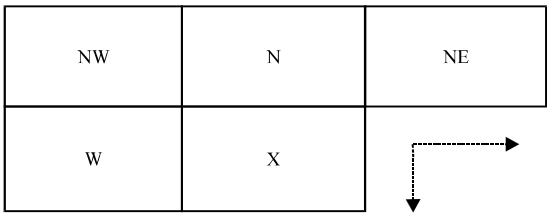


Fig. 3: Pixel prediction template

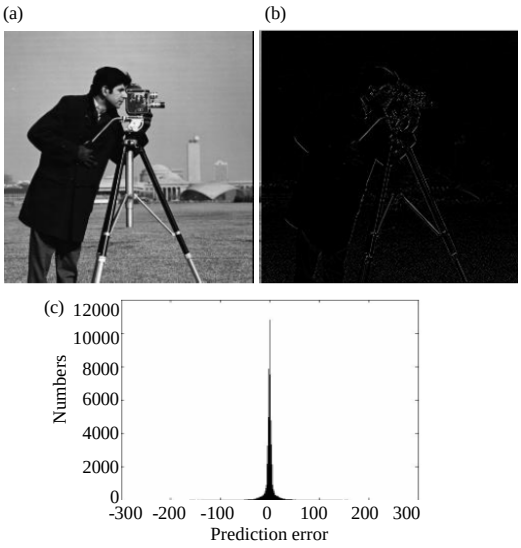


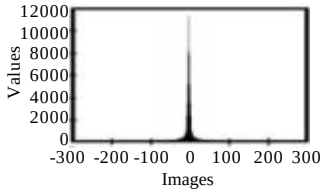
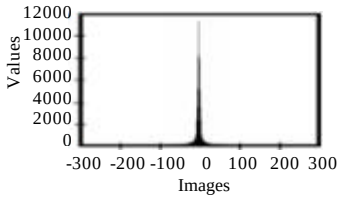
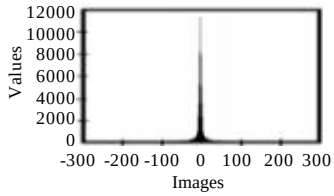
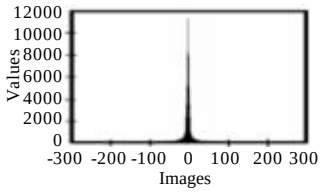
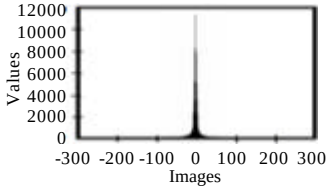
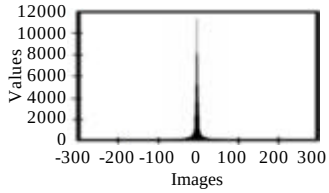
Fig. 4: a) Cameraman image; b) Prediction error image and c) Prediction error histogram

$$x = \begin{cases} \min(N, W) & \text{if } NM = \max(W, NW, N) \\ \max(N, W) & \text{if } NM = \min(W, NW, N) \\ W\text{-}NW+N & \text{else} \end{cases} \quad (3)$$

The prediction for the first pixel is zero. The rest of the first row pixels are predicted by the value to the left and the rest of first column by the value to the top. The prediction error histogram H is formed with bins representing values from -255 to 255.

The following predictors are considered in this research. The mean predictor uses the arithmetic mean of W, NW and N values. The median predictor uses the median of these values. The W, NW and N values themselves can be used as the prediction for x. These predictors except the median are used in JPEG-LS scheme. The predictors are listed in Table 1. The data in Table 1 demonstrates the superior performance of JPEG-LS nonlinear predictor for cameraman image. However, the performance of the predictors will be analyzed in more detail for more images later (Fig. 4).

Table 1: Predictors and the characteristics of prediction errors of cameraman image

Preditors	X	Error histogram (cameraman image)	Mean of squared (cameraman image)	No. of zero values (cameraman image)
JPEG-LS	Equation 3		215.71	10786
Mean	W+NW+N/3		350.88	3720
Median	median (W, NW, N)		508.21	9889
NW	NW		708.53	8073
W	W		517.38	10163
N	N		316.68	9469

The secret image pixel values are converted to trits (digits of base 3). The pixel intensity values are in the range 0-255 and can be captured in 6 trits. The least significant trit (lst) is ignored and the next least significant trit adjusted leading to a maximum error of ±1. For example the pixel intensity value of 2 when converted to trits gives [0, 0, 0, 0, 0, 2]. After removing the sixth trit, the fifth trit is changed to 1 to give [0, 0, 0, 0, 1]. For the pixel intensity values of 0-255, five trits per pixel can always capture the values with error bounded by ±1. This corresponds to expected mean squared error of 0.6641 and expected PSNR of 49.9 dB. The 5 trits per pixel are then embedded into the prediction error histogram of the cover image. The histogram bins to the left of the peak bin of zero are shifted to the left and the bins to the right of the peak bin are shifted to the right. This creates two empty bins at

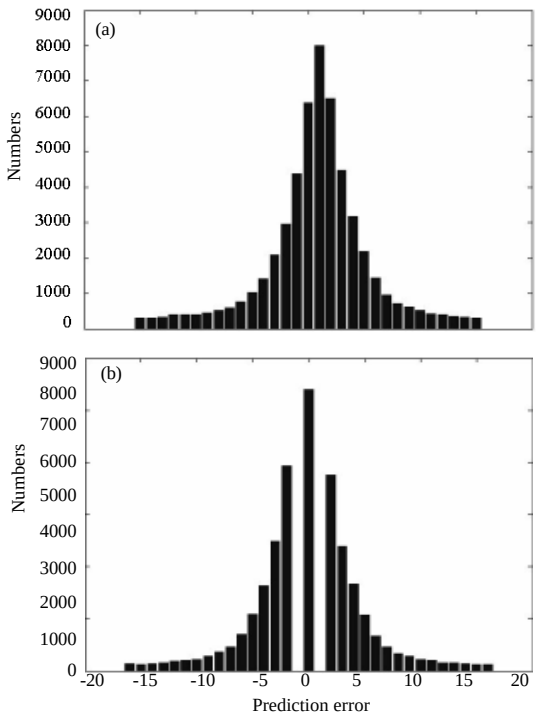


Fig. 5: Proposed prediction error histogram modification illustration; a) Original historam and b) Mondified histogram

either sides of the peak bin as shown in Fig. 5. The values of the peak bin are visited in the pixel order and shifted to the left, left unperturbed or shifted to the right depending on the three possible values of the trit. The modified error values are added to the prediction image to get the stego image which is then transmitted to the receiver. Security is enhanced by permuting the pixel order according to a shared key K, known only to the sender and receiver. It is assumed that the key K is shared through a secure channel before the steganographic exchange. The embedding equation is:

$$p = \begin{cases} p\text{-}1, & m = 0 \\ p, & m = 1 \\ p\text{+}1, & m = 2 \end{cases} \quad (4)$$

where p is the actual prediction error and p' is the perturbed error after embedding message trit m. At the extraction stage the bins are brought closer towards the zero bin with values of -1 and +1 becoming zero. The perturbed errors are added to the prediction image to get the stego image.The proposed technique is named as pixel trinary prediction error method. The block diagrams for embedding and extraction procedures are show in Fig. 6 and 7.

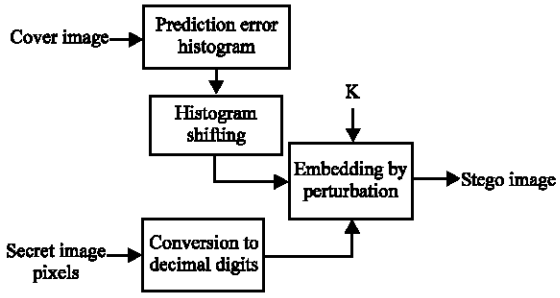


Fig. 6: Bolck diagram of embedding

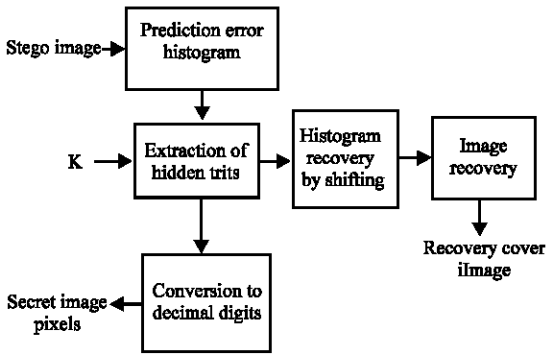


Fig. 7: Bolck diagram of extraction

Steps of embedding:

- The predictor is applied on the cover image and the prediction error histogram is calculated
- The histogram bins to the left and right of the zero bin are shifted to the left and right respectively
- The secret image is resized so that the number of trits is equal to the number of zero bin values of the cover image
- The trits are embedded in the prediction error histogram by visiting the pixels in a pseudorandom order seeded by the secret shared key K using Eq. 4
- The modified stego prediction errors are added to the predicted values and the values are clipped to the range [0, 255] to get the stego image

Steps of extraction:

- The predictor is applied to the stego image and the prediction errors are calculated
- The prediction errors of values in the set {-1, 0, 1} are visited in the pseudorandom order seeded by the secret shared key K and the trit values are read
- The trits are converted back to the secret image
- The prediction errors of values {-1, 1} are restored to value 0. All other values are restored by bringing them closer to zero by one

- The restored prediction errors are added back to the prediction image to get the restored cover image

Resizing the secret image: n order to fit the secret image into the cover pixels, the secret image is reduced in size so that the number of trits is equal to the number of zero values in the prediction error image. If the size of secret image is m×n and the number of zero prediction errors in cover image is z then the new size of the secret image is given by:

$$\text{Size} = \text{floor}\left(\text{sqrt}\left(\frac{zm}{n}\right)\right)\times\text{floor}\left(\text{sqrt}\left(\frac{zm}{n}\right)\right)\frac{n}{m} \quad (5)$$

RESULTS AND DISCUSSION

The proposed method was implemented in MATLAB 2013. A dataset containing several test images obtained from [22] is used. The secret image used was taken from Yale B face dataset (Lee *et at.*, 2005). All cover images are of size 256×256. Figure 8 shows the original and recovered secret image and its trits representation.

The cover and stego images are illustrated with their prediction error image in Fig. 9-12. The original and perturbed prediction error histogram are also shown. Among the images used for illustration, Lena is a color image. The embedding is done by modifying the prediction error histogram with values from all three color planes. The security key K is used to randomize the visiting order of the zero error pixels locations. It is assumed that the secret key K is shared among the sender and receiver through a secure channel besides the image steganography. The means of steganographic exchange of keys is out of scope of this work. The embedding can be repeated several times as long as there are sufficient zero prediction error pixels in the cover image.

Table 2 displays the embedding capacity in different images and for different predictors used. It can be seen that JPEG-LS nonlinear predictor offers the most zero errors and thus the highest embedding capacity. The visual distortion of the stego images are measured using Mean Squared Error (MSE), Peak Signal to Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM). While MSE and PSNR are based on the difference in raw values of individual pixels, SSIM is a more advanced similarity measure based on edges. The high values of all the performance indicators show that subjective detection of hidden data is impossible in the proposed method. The relatively poor performance on Mandril and Walkbridge images can be explained by the presence of large amounts of texture in the images. In general images with smooth regions with zero prediction errors offer more embedding capacity (Table 3).

Table 2: Embedding capacity (kB) for different images and predictors

Images	JPEG-LS nonlinear	Mean	Median	NW	W	N
Cameraman	2.11	0.73	1.93	1.58	1.99	1.85
Lake	1.49	0.46	1.21	0.97	1.24	1.18
Lena (color)	4.11	0.55	1.51	1.19	1.29	1.82
Pirate	1.44	0.44	1.31	0.90	1.08	1.30
Living room	1.47	0.31	0.83	0.65	0.98	1.18
Mandril	0.52	0.16	0.42	0.34	0.42	0.43
Peppers	1.75	0.53	1.46	1.15	1.44	1.51
Walkbridge	0.70	0.16	0.45	0.34	0.62	0.47

Table 3: Stego image distortion

Images	MSE	PSNR (dB)	SSIM
Cameraman	0.9450	48.38	0.9925
Lake	0.9591	48.31	0.9954
Lena (color)	0.9648	48.29	0.9939
Pirate	0.9632	48.29	0.9959
Living room	0.9606	48.31	0.9962
Mandril	0.9854	48.19	0.9976
Peppers	0.9544	48.33	0.9941
Walkbridge	0.9807	48.22	0.9980

Table 4: Time taken for embedding

Images	Execution time
Cameraman	0.80
Lake	0.78
Lena (color)	1.08
Pirate	0.78
Living room	0.78
Mandril	0.78
Peppers	0.77
Walkbridge	0.81

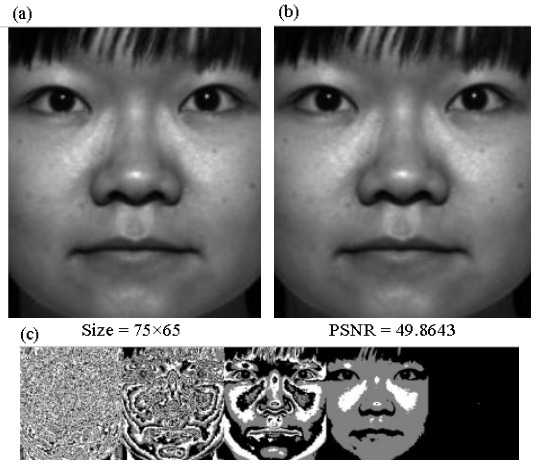


Fig. 8: a) Original secret image at the sender; b) Recovered secret image by the receiver and c) Visualization of the trits representation

Figure 13 shows the Peak Signal to Noise Ratio values with increasing embedding capacity by repeated embedding. The PSNR values are very high and above 35 for 1 bit per pixel embedding. This shows that the proposed method is suitable for high capacity data embedding. The proposed method is compared against several methods in literature in Fig. 14. The PSNR values of the proposed method are the

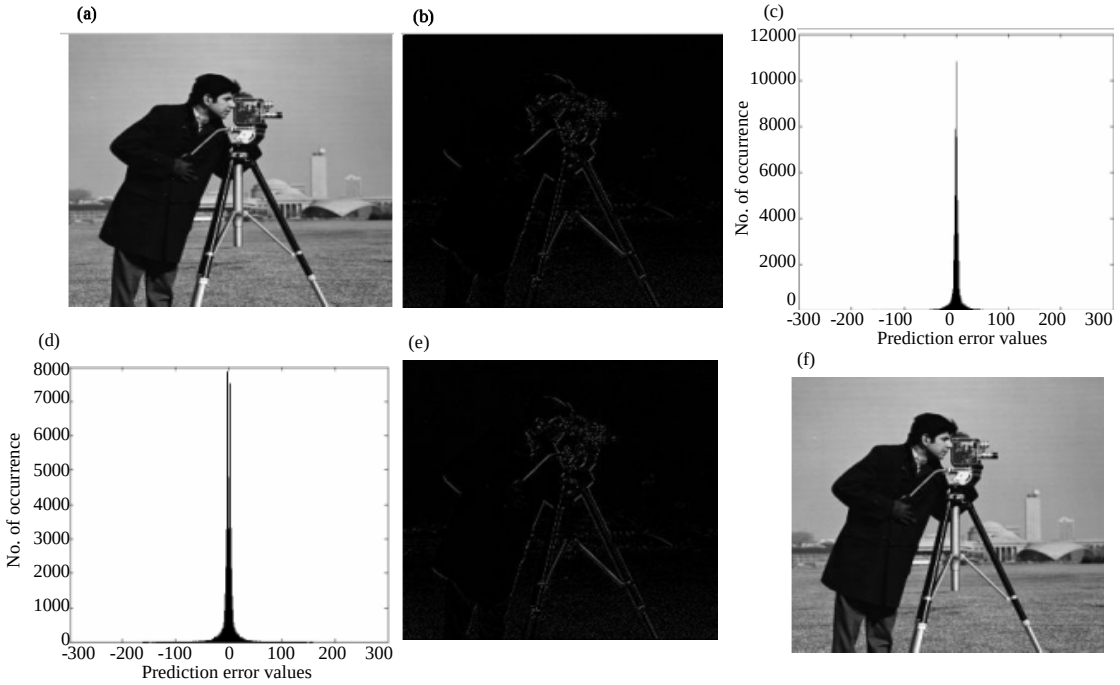


Fig. 9: Camera man image; a) Original image; b) Prediction error ; c) PE histogram; d) PE histogram after embedding; e) Embedded prediction error image and f) Stego image

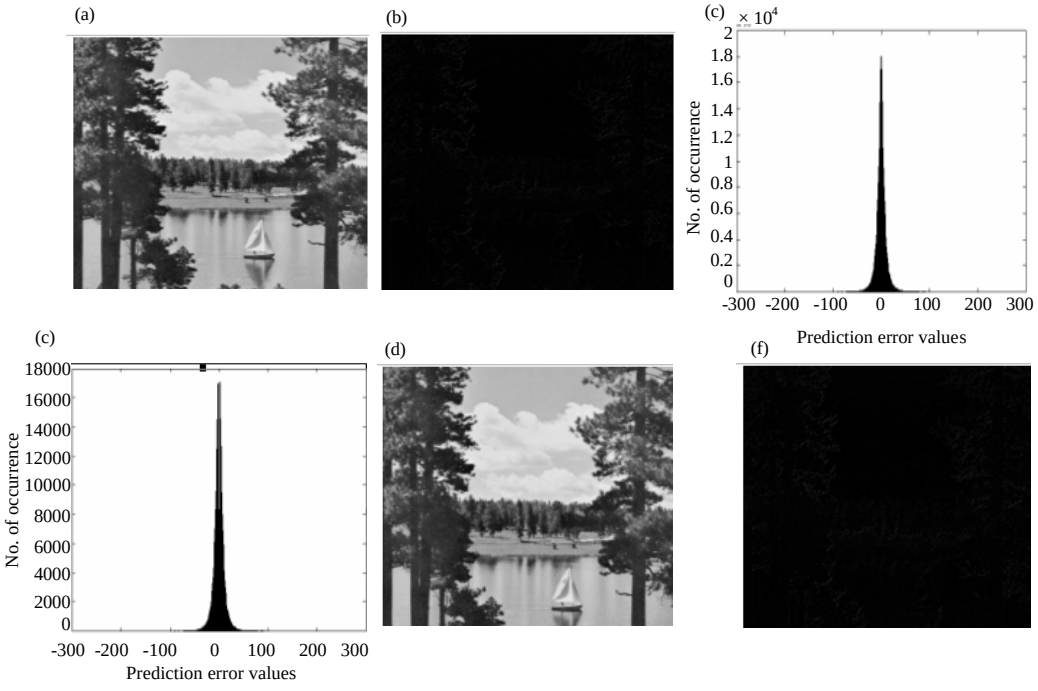


Fig. 10: Lake image; a) Original image; b) Prediction error; c) PE histogram; d) PE histogram after embedding; e) Embedded prediction error image and f) Stego image

highest except (Lee *et at*., 2007) which offers embedding capacity of <0.2 bpp. The other methods compared are (Ni *et al*., 2006; Alattar, 2004; Fallahpour and Sedaaghi, 2007; Celik *et al*., 2002; Tsai *et al*., 2009).

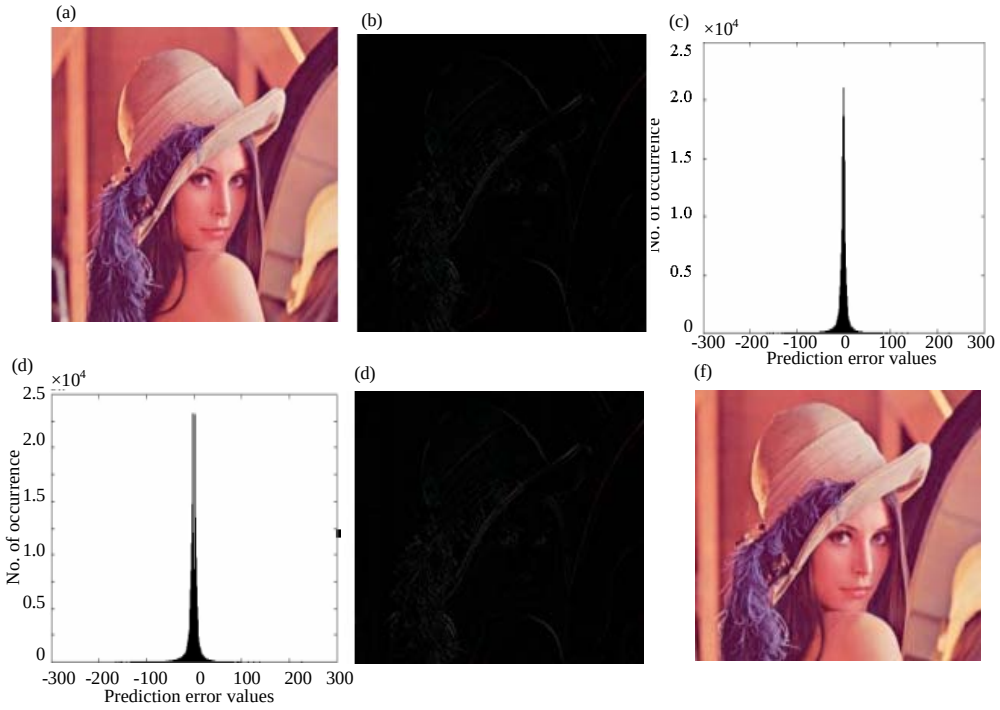


Fig. 11: Lena image; a) Original image; b) Prediction error; c) PE histogram; d) PE histogram after embedding; e) Embedded prediction error image and f) Stego image

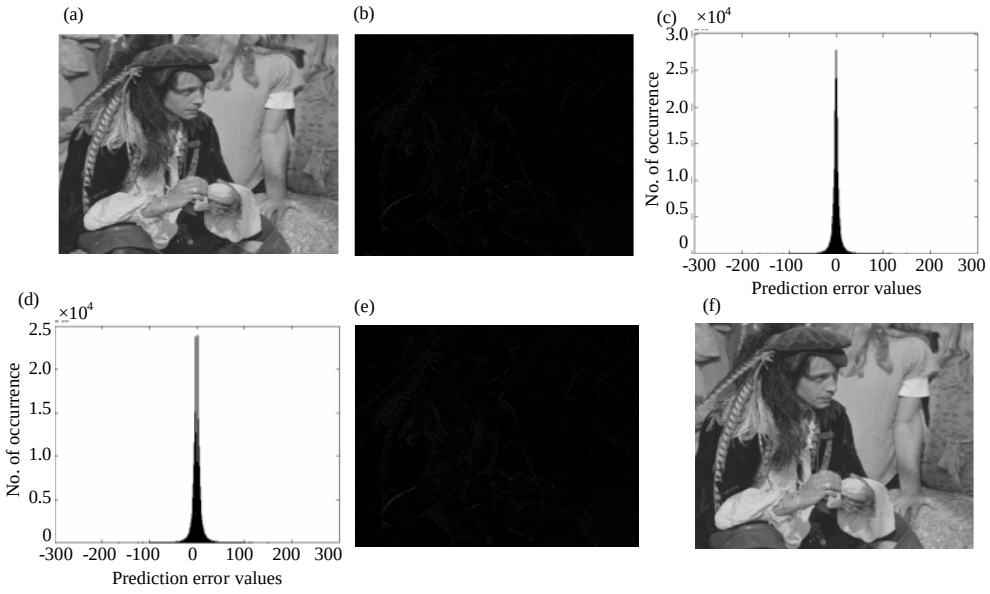


Fig. 12: Pirate Image; a) Original image; b) Prediction error; c) PE histogram; d) PE histogram after embedding; e) Embedded prediction error image and f) Stego image

The proposed method is computationally efficient. The MATLAB implementation was run on a laptop with iCore5 processor and 4GB RAM. The embedding and extraction took less than a second each. Table 4 gives the execution time for embedding the test images with the chosen secret image.

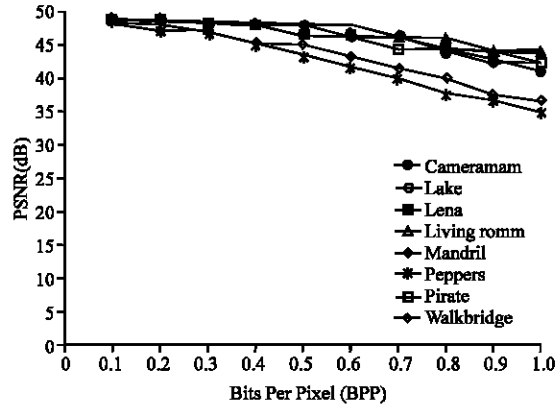


Fig. 13: PSNR vs. embedding capacity for the test images

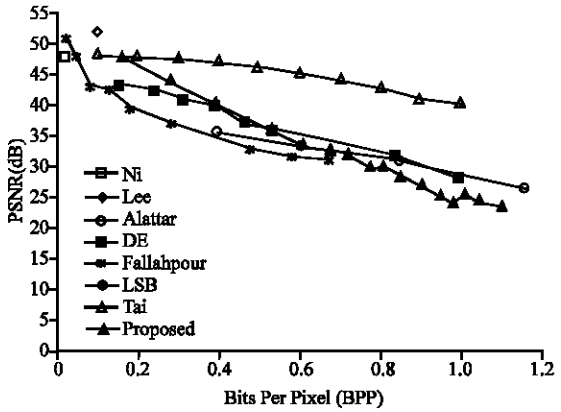


Fig. 14: Relative performance of proposed method

CONCLUSION

A method for reversible data hiding in images is presented. The payload is a secret image which is resized, converted to trinary representation and embedded. The prediction error histogram is used as the carrier. The values are shifted to create empty bins to the either side of the peak bin at value zero. The trits are encoded by perturbing the zero values. Additional embedding can be done by repeating the procedure a few more times. Experimental results indicated that the proposed method has high embedding capacity and gives a better visual distortion performance of above 40 dB. The method is suitable for reversible embedding of secret image within another image. Comparatively the proposed method performs better than other reversible data hiding methods in literature.

RECOMMENDATIONS

Future research efforts can explore embedding in higher bases. Three dimensional color image histograms can also be explored for embedding.

REFERENCES

Alattar, A.M., 2004. Reversible watermark using the difference expansion of a generalized integer transform. IEEE Trans. Image Process., 13: 1147-1156.

Bender, W., D. Gruhl, N. Morimoto and A. Lu, 1996. Techniques for data hiding. IBM Syst. J., 35: 313-336.

Celik, M.U., G. Sharma, A.M. Tekalp and E. Saber, 2002. Reversible data hiding. Processing of the International Conference on Image Processing Vol. 2, September 22-25, 2002, IEEE, Rochester, New York, USA., ISBN: 0-7803-7622-6, pp: 157-160.

Celik, M.U., G. Sharma, A.M. Tekalp and E. Saber, 2005. Lossless generalized-LSB data embedding. IEEE Trans. Image Process., 14: 253-266.

Chen, C.C. and C.C. Chang, 2010. High capacity SMVQ-based hiding scheme using adaptive index. Signal Process., 90: 2141-2149.

Fallahpour, M. and M.H. Sedaaghi, 2007. High capacity lossless data hiding based on histogram modification. IEICE Electr. Express, 4: 205-210.

Fridrich, J., M. Goljan and R. Du, 2001. Invertible authentication. Proceedings of SPIE Security and Watermarking of Multimedia Contents, Volume 4314, January 21-26, 2001, San Jose, CA., USA., pp: 197-208.

Gkizeli, M., D.A. Pados and M.J. Medley, 2007. Optimal signature design for spread-spectrum steganography. IEEE. Trans. Image Process., 16: 391-405.

Goljan, M., J. Fridrich and R. Du, 2001. Distortion-Free Data Embedding for Images. In: Information Hiding, Moskowitz, I.S. (Ed.). Springer, Berlin, Germany, ISBN:978-3-540-42733-9, pp: 27-41.

Guillemot, L. and J.M. Moureaux, 2010. Modulated lattice vector quantization: How to make quantization index modulation an efficient variable rate source coder. IEEE. Trans. Commun., 58: 3165-3174.

Hong, W., T.S. Chen and H.Y. Wu, 2012. An improved reversible data hiding in encrypted images using side match. IEEE. Signal Process. Lett., 19: 199-202.

Honsinger, C.W., P.W. Jones, M. Rabbani and J.C. Stoffel, 2001. Lossless recovery of an original image containing embedded data. U.S. Patent No. 6,278,791, August 21, 2001, Washington, DC.

Hsu, F.H., M.H. Wu, C.H. Yang and S.J. Wang, 2014. Visible watermarking with reversibility of multimedia images for ownership declarations. J. Supercomputing, 70: 247-268.

Kalantari, N.K and S.M. Ahadi, 2010. A logarithmic quantization index modulation for perceptually better data hiding. Trans. Image Process., 19: 1504-1517.

Lee, C.F., C.C. Chang, J.J. Li and Y.H. Wu, 2016. A survey of reversible data hiding schemes based on pixel value ordering. Proceedings of the International Conference on Nicograph (NicoInt), July 6-8, 2016, IEEE, Hanzhou, China, ISBN:978-1-5090-2305-9, pp: 68-74.

Lee, K.C., J. Ho and D.J. Kriegman, 2005. Acquiring linear subspaces for face recognition under variable lighting. IEEE. Trans. Pattern Anal. Mach. Intell., 27: 684-698.

Lee, S., C.D. Yoo and T. Kalker, 2007. Reversible image watermarking based on integer-to-integer wavelet trans form. IEEE. Trans. Inf. Forensics Secur., 2: 321-330.

Ni, Z., Y.Q. Shi, N. Ansari and W. Su, 2006. Reversible data hiding. IEEE. Trans. Circuits Syst. Video Technol., 16: 354-362.

Provos, N. and P. Honeyman, 2003. Hide and seek: An introduction to steganography. IEEE Secur. Privacy, 1: 32-44.

Sarreshtedari, S. and M.A. Akhaee, 2013. One-third probability embedding: A new±1 histogram compensating image least significant bit steganography scheme. IET. Image Process., 8: 78-89.

Tai, W.L., C.M. Yeh and C.C. Chang, 2009. Reversible data hiding based on histogram modification of pixel differences. IEEE Trans. Circuits Syst. Video Technol., 19: 906-910.

Tian, J., 2003. Reversible data embedding using a difference expansion. IEEE Trans. Circ. Syst. Video Technol., 13: 890-896.

Tsai, P.Y., Y.C. Hu and H.L. Yeh, 2009. Reversible image hiding scheme using predictive coding and histogram shifting. Signal Process., 89: 1129-1143.

Wu, M., H. Yu and B. Liu, 2003. Data hiding in image and video. II. designs and applications. IEEE. Trans. Image Process., 12: 696-705.

Xuan, G., J. Zhu, J. Chen, Y.Q. Shi and Z. Ni *et al*., 2002. Distortionless data hiding based on integer wavelet transform. Electron. Lett., 38: 1646-1648.

Zhang, X., 2012. Separable reversible data hiding in encrypted image. Trans. Inform. Forensics Secur., 7: 826-832.