IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 5, May 2017

Reversible Image Data Hiding with Contrast Enhancement

Prof. M.S. Sadavarte[1], **Pratima B. Choudhari**[2]

Assistant Professor, Electronics and Telecommunication, Government College of Engineering, Jalgaon, India[1]

Student, Electronics and Telecommunication, Government College of Engineering, Jalgaon, India[2]

Abstract: In this report, reversible data hiding (RDH) algorithm is proposed for digital images. Instead of trying to keep the PSNR value high, the proposed algorithm enhances the contrast of a host image to improve its visual quality. The highest two bins in the histogram are selected for data embedding so that histogram equalization can be performed by repeating the process. The side information is embedded along with the message bits into the host image so that the original image is completely recoverable. The proposed algorithm is used for two sets of images to demonstrate its efficiency. To our best knowledge, it is the first algorithm that achieves image contrast enhancement by RDH. Furthermore, the evaluation results show that the visual quality can be preserved after a considerable amount of message bits have been embedded into the contrast-enhanced images, even better than three specific MATLAB functions used for image contrast enhancement.

Keywords: RDH, MATLAB, Image, Histogram.

I. INTRODUCTION

Reversible Data Hiding (RDH) has been intensively studied in the community of signal processing. also referred as invertible or lossless data hiding, RDH is to embed a piece of information into a host signal to generate the marked one, from which the original signal can be exactly recovered after extracting the embedded data. the technique of RDH is useful in some sensitive applications where no permanent change is allowed on the host signal. in the literature, most of the proposed algorithms are for digital images to embed invisible data or a visible watermark.

To evaluate the performance of a RDH algorithm, the hiding rate and the marked image quality are important metrics. there exists a trade-off between them because increasing the hiding rate often causes more distortion in image content. to measure the distortion, the peak signal-to-noise ratio (PSNR) value of modification of image histogram provides less embedding capacity. in contrast, the more recent algorithms manipulate the more centrally distributed prediction errors by exploiting the correlations between neighboring pixels so that less distortion is caused by data hiding. although the PSNR of a marked image generated with a prediction error based algorithm is kept high, the visual quality can hardly be improved because more or less distortion has been introduced by the embedding operations. for the images acquired with poor illumination, improving the visual quality is more important than keeping the PSNR value high. moreover, contrast enhancement of medical or satellite images is desired to show the details for visual inspection. although the psnr value of the enhanced image is often low, the visibility of image details has been improved. to our best knowledge, there is no existing RDH algorithm that performs the task of contrast enhancement so as to improve the visual quality of host images. So in this study, we aim at inventing a new RDH algorithm to achieve the property of contrast enhancement instead of just keeping the PSNR value high.

In principle, image contrast enhancement can be achieved by histogram equalization. to perform data embedding and contrast enhancement at the same time, the proposed algorithm is performed by modifying the histogram of pixel values. Firstly, the two peaks (i.e. the highest two bins) in the histogram are found out. 2 the bins between the peaks are unchanged while the outer bins are shifted outward so that each of the two peaks can be split into two adjacent bins. to increase the embedding capacity, the highest two bins in the modified histogram can be further chosen to be split, and so on until satisfactory contrast enhancement effect is achieved. To avoid the overflows and underflows due to histogram modification, the bounding pixel values are pre-processed and a location map is generated to memorize their locations. for the recovery of the original image, the location map is embedded into the host image, together with the message bits and other side information. so blind data extraction and complete recovery of the original image are both enabled. the proposed algorithm was applied to two set of images to demonstrate its efficiency. to our best knowledge, it is the first algorithm that achieves image contrast enhancement by RDH. Furthermore, the evaluation results show that the visual quality can be preserved after a considerable amount of message bits have been embedded into the contrast-enhanced images, even better than three specific MATLAB functions used for image contrast enhancement.

DOI10.17148/IJARCCE.2017.6576

II. OBJECTIVE

A. The main objective of the “reversible image data hiding with contrast enhancement” is to achieve real reversibility, separate data extraction and great improvement on the quality of marked images.
B. To increase security while communicating over the untrusted media such as internet.
C. RDH is the technique that used in securely transfers the information with the use of algorithm which is un-readable by the third-party.

III.SYSTEM DEVELOPMENNT

Data embedding in the reversible manner which is the data embedding without any loss embeds the data or payload into digital image in reversible manner. After data embedding the quality of original image may be degraded which is to be avoided. The attractive property of data embedding in reversible manner is reversibility, which is after data extraction the original quality image is restored back. Reversible data embedding hides some information in a digital image in such a way that an approved party image to its original state. The presentation of a reversible data-embedding algorithm can be measured using following,
1. Data embedding capacity limit
2. Visual quality
3. Complexity

The data without any distortion embedding is the attractive feature of reversible data embedding. Data will certainly change the original content by embedding some data into it. Even a very slight change in pixel values may not be pleasing, particularly in military data and medical data. In such a circumstances, every small part of information is important. From the application point of view, since the differentiation between the implanted image and original image is almost discernible from human eyes, reversible data implanting could be thought as a top secret communication channel since reversible data implanting can be used as an information transporter. Mainly the data hiding techniques are classified into two techniques:
1. Reversible data hiding technique
2. Irreversible data hiding technique
1. Reversible data hiding technique: In this technique the message signal as well as the original cover can be with no loss recovered simultaneously.
2. Irreversible data hiding technique: In this technique the message signal can be recovered with no loss but the original cover can be lost. So in general reversible data hiding techniques can be used now a days.

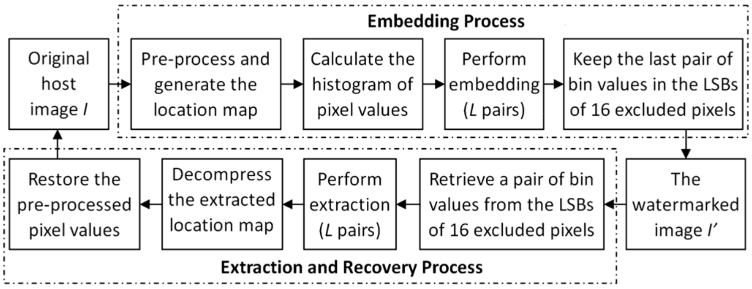
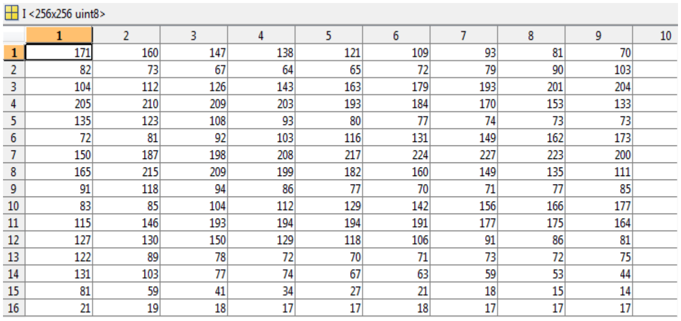

Embedding Process
Original host image I
Pre-process and generate the location map
Calculate the histogram of pixel values
Perform embedding (L pairs)
Keep the last pair of bin values in the LSBs of 16 excluded pixels
The watermarked image I′
Retrieve a pair of bin values from the LSBs of 16 excluded pixels
Perform extraction (L pairs)
Decompress the extracted location map
Restore the pre-processed pixel values
Extraction and Recovery Process

Fig. 1. Procedure of the proposed RDH algorithm

When we read any image in matlab then it gives matrix of pixel values as follows



I <256x256 uint8>

	1	2	3	4	5	6	7	8	9	10
1	171	160	147	138	121	109	93	81	70	
2	82	73	67	64	65	72	79	90	103	
3	104	112	126	143	163	179	193	201	204	
4	205	210	209	203	193	184	170	153	133	
5	135	123	108	93	80	77	74	73	73	
6	72	81	92	103	116	131	149	162	173	
7	150	187	198	208	217	224	227	223	200	
8	165	215	209	199	182	160	149	135	111	
9	91	118	94	86	77	70	71	77	85	
10	83	85	104	112	129	142	156	166	177	
11	115	146	193	194	194	191	177	175	164	
12	127	130	150	129	118	106	91	86	81	
13	122	89	78	72	70	71	73	72	75	
14	131	103	77	74	67	63	59	53	44	
15	81	59	41	34	27	21	18	15	14	
16	21	19	18	17	17	18	17	17	17	

Fig 2. Image in Matrix Form

Total no. of pixels in image = no. of rows * no.of columns= 256 *256
= 65536 pixels
Intensity value is calculated from the following gray scale. Where, 0 = Black , 255 = White

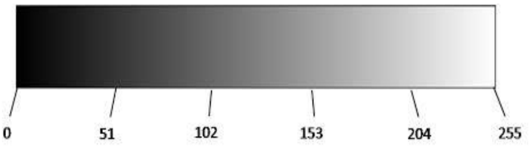


Fig 3. Gray scale

A. Data Embedding by Histogram Modification

The algorithm to be presented is primarily for gray-level images but can be easily extended to color images. Given an 8-bit gray-level image I, the image histogram can be calculated by counting the pixels with a gray-level value j for j € { j=0,1,….254,255 } . We use $h_I$to denote the image histogram so that represents h_I(j) the number of pixels with a value j. Suppose I consists N of different pixel values. Then there are N nonempty bins inh_I , from which the two peaks (i.e. the highest two bins) are chosen and the corresponding smaller and bigger values are denoted by $I_S$and I_R, respectively. For a pixel counted in with value i , data embedding is performed

$$i' = \begin{cases} i-1, & \text{for } i < I_S \\ I_S - b_k, & \text{for } i = I_S \\ i, & \text{for } I_S < i < I_R \\ I_R + b_k, & \text{for } i = I_R \\ i+1, & \text{for } i > I_R, \end{cases} \qquad (1)$$

Histogram plots how many times each intensity value in image occurs**.** It plots the number of pixels for each intensity value.

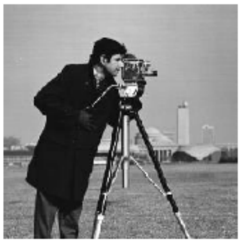


Fig 4.Original image

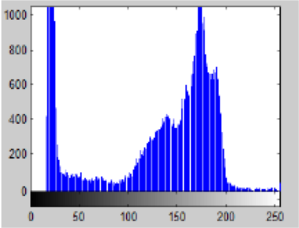


Fig 5. Histogram

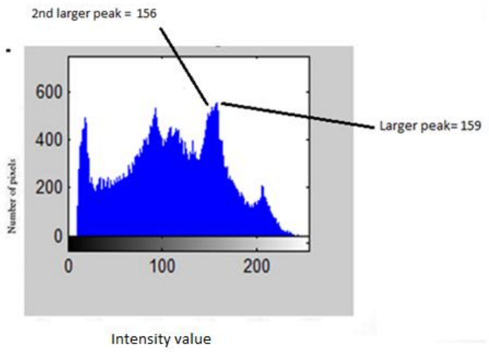


Fig 6. Histogram

Larger peak is denoted by $I_{R.}$, let $I_{R.}$ =159 and 2nd larger peak is denoted by I_s , let I_s =156

where i is the modified pixel value, and is the b_k is the k-th message bit (0 or 1) to be hidden. By applying Eq. (1) to every pixel counted in h_I, totally $h_I(I_S) + h_I(I_R)$ binary values are embedded. Given that there is no bounding value (0 or 255) in I (otherwise pre-process is needed), there will be N+2 bins in the modified histogram. That is, the bins between the two peaks are unchanged while the outer ones are shifted outward so that each of the peaks can be split into two adjacent bins (i.e. I_S-1 and I_S, $I_R$and I_R+1, respectively). Image scanning in matlab is as follows

DOI10.17148/IJARCCE.2017.6576

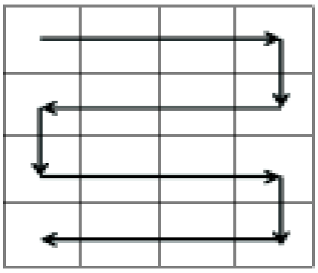


Fig 7. Inverse s order scan

Data which is to be embedded is The peak values I_s and $I_{R.}$ need to be provided to extract embedded data.We are going to take no. of paires of I_s and $I_{R.}$ to increase embedding capacity. And Keyword for image security

The peak values and need to be provided to extract the embedded data. One way to keep them is to exclude 16 pixels in from histogram computing. The least significant bits (LSB) of those pixels are collected and included in the binary values to be hidden. After applying Eq. (1) to each pixel counted in for data embedding, the values ofI_S and I_R(each with 8 bits) are used to replace the LSBs of the 16 excluded pixels by bitwise operation. To extract the embedded data, the peak values need to be retrieved and the histogram of the marked image I is calculated excluding the 16 pixels aforementioned. Then the following operation is performed on any pixel counted in the histogram and with the value of I_S-1, I_S, $I_R$and I_R+1.

$$b'_k = \begin{cases} 1, & \text{if } i' = I_S - 1 \\ 0, & \text{if } i' = I_S \\ 0, & \text{if } i' = I_R \\ 1, & \text{if } i' = I_R + 1, \end{cases} \qquad (2)$$

Where $b_K$is the k-th binary value extracted from the marked image I . The extraction operations are performed in the same order as that of the embedding operations. According to Eq. (1), the following operation is performed on every pixel counted in the histogram to recover its original value:

$$i = \begin{cases} i' + 1, & \text{for } i' < I_S - 1 \\ I_S, & \text{for } i' = I_S - 1 \text{ or } i' = I_S \\ I_R, & \text{for } i' = I_R \text{ or } i' = I_R + 1 \\ i' - 1, & \text{for } i' > I_R + 1 \end{cases}. \qquad (3)$$

The original LSBs of 16 excluded pixels are obtained from the extracted binary values. The excluded pixels can be restored by writing them back so as to recover the original image.

B. Pre-Process for Complete Recovery

In the aforementioned algorithm, it is required that all pixels counted in $h_I$are within {1,……, 254}. If there is any boundingpixel value (0 or 255), overflow or underflow will be caused by histogram shifting. To avoid it, the histogram needs to be pre-processed prior to the histogram modification operations. Specifically, the pixel values of 0 and 255 are modified to 1 and 254, respectively. Therefore, no overflow or underflow will be caused because the possible change of each pixel value is +1 or -1 . To memorize the pre-processed pixels, a location map with thesame size as the original image is generated by assigning 1 to the location of a modified pixel, and 0 to that of an unchanged one (including the 16 excluded pixels). The locationmap can be pre-computed and included into the binary values to be hidden. In the extraction and recovery process, it can be obtained from the data extracted from themarked image so that the pixelsmodified in the pre-process can be identified. By restoring the original values of those pixels accordingly, the original image can be completely recovered

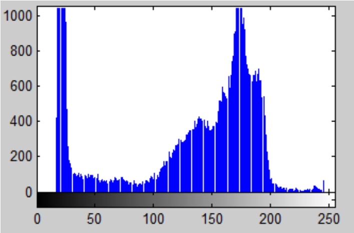


Fig 8. Shifted Histogram

The main advantage of the histogram shifting is it removes the underflow and overflow. During the addition of watermark to an original image caution must be taken to prevent the Negative or Gray level values called underflow and positive values called overflows.. Drawbacks of the Existing system is that they areLess Efficient, Causes Overflow and Underflow and result is having More Distortions.

A. Contrast Enhancement

Contrast Enhancement is the process of changing the intensity of the pixels of the input image in order to utilize maximum possible bins. Histogram equalization is one of the method of achieving image contrast enhancement, it is performed by modifying the histogram of pixel values. The two highest bins of the histogram are found out and then the bins between the peak values are kept unchanged whereas the outer bins are shifted outwards thereby dividing the two peaks in two adjacent bins. This process is used to obtain data embedding with contrast enhancement simultaneously. The highest two bins in the histogram can be further split and so on to achieve increased embedding capacity along with satisfactory contrast enhancement. To recover original image, the location map is fixed into the cover image with the bits of message and other additional information.

Histogram equalization is a technique for adjusting image intensities to enhance contrast.(equal distribution of intensities).As we are adding or subtracting data from the intensities I_s and $I_{R.}$ we get two more peaks $I_s -1$ and $I_{R.}+1$. In this way data embedding and contrast enhancement are simultaneously performed.

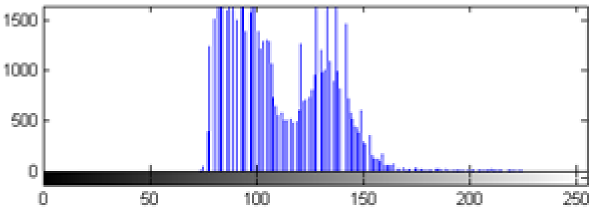


Fig 9. Original Histogram

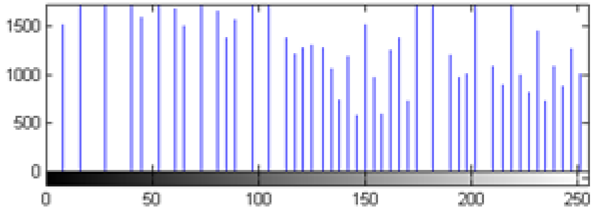


Fig. 10 Equalized Histogram

Each of the two peaks in the histogram is split into two adjacent bins with the similar or same heights because the numbers of 0s and 1s in the message bits are required to be almost equal. To increase the hiding rate, the highest two bins in the modified histogram are further chosen to be split by applying Eq. (1) to all pixels counted in the histogram. The same process can be repeated by splitting each of the two peaks into two adjacent bins with the similar heights to achieve the histogram equalization effect. In this way, data embedding and contrast enhancement are simultaneously performed. Given that the pair number of the histogram peaks to be split is L, the range of pixel values from 0 to L-1 are added by L while the pixels from 256-L to 255 are subtracted by L in the pre-process (noting L is a positive integer). A location map is generated by assigning 1s to the modified pixels, and 0s to the others.

The location map can be pre-computed and compressed to be firstly embedded into the host image. The value of L, the size of the compressed location map, and the previous peak values, in contrary, are embedded with the last two peaks to be split, whose values are stored in the LSBs of the 16 excluded pixels. In the extraction process, the last split peak values are retrieved and the data embedded with them are extracted with Eq. (2). After restoring the histogram with Eq. (3), the data embedded with the previously split peaks can also be extracted by processing them pair by pair. At last, the location map is obtained from the extracted data to identify the pixel values modified in the pre-process.

B. Procedure of the Proposed Algorithm

The procedure of the proposed algorithm is illustrated in Fig. 1. Given that totally L pairs of histogram bins are to be split for data embedding, the **embedding** procedure includes the following steps:

1) Pre-process: The pixels in the range of [0 to L-1]and are processed as mentioned above excluding the first 16 pixels in the bottom row. A location map is generated to record the locations of those pixels and compressed by the JBIG2 standard to reduce its length.

DOI10.17148/IJARCCE.2017.6576

2) The image histogram is calculated without counting the first 16 pixels in the bottom row.
3) Embedding: The two peaks (i.e. the highest two bins) in the histogram are split for data embedding by applying Eq. (1) to every pixel counted in the histogram. Then the two peaks in the modified histogram are chosen to be split, and so on until L pairs are split. The bit stream of the compressed location map is embedded before the message bits (binary values). The value of L, the length of the compressed location map, the LSBs collected from the 16 excluded pixels, and the previous peak values are embedded with the last two peaks to be split.
4) The lastly split peak values are used to replace the LSBs of the 16 excluded pixels to form the marked image.
The data **extraction** and original image **recovery** process include the following steps:
1)The LSBs of the 16 excluded pixels are retrieved so that the values of the last two split peaks are known.
2)The data embedded with the last two split peaks are extracted by using Eq. (2) so that the value of L, the length of the compressed location map, the original LSBs of 16
excluded pixels, and the previously split peak values are known. Then the recovery operations are carried out by processing all pixels except the 16 excluded ones with Eq. The process of extraction and recovery is repeated until all of the split peaks are restored and the data embedded with them are extracted.
3)The compressed location map is obtained from the extracted binary values and decompressed to the original size. With the decompressed map, those pixels modified in preprocess are identified. Among them, a pixel value is subtracted by L if it is less than 128, or increased by otherwise. To comply with this rule, the maximum value of L is 64 to avoid ambiguity. At last, the original image isrecovered by writing back the original LSBs of 16 excluded pixels.

REFERENCES

[1] Kede Ma. Weiming Zhang, Xianfeng Zhao, Nenghai Yu,Fenghua Li, " Reversible Data Hiding in Images by Reserving Room Before Encryption", IEEE Trans on Information Forensics and security, Vol. 8, No. 3, March 2013
[2] Wen Chung Kuo, Po Yu Lai, Lih Chyau Wuu, "Adaptive Reversible Data Hiding Based on Histogram", 10th International Conference on Intelligent Systems Design and Application, l' IEEE 2010 (2002) The IEEE website. [Online]. Available: http://www.ieee.org/
[3] Kuo-Ming Hung, Wen-Kai Su, Ting-Wen Chen, Li-Ming Chen, "Reversible Data Hiding Base on VQ and Halftoning Technique", M. Wegmuller, J. P. von der Weid, P. Oberson, and N. Gisin, “High International Conference on Microelectronics, Communication and Renewable Energy (ICMiCR-2013).
[4] Jose, R.; Abraham, G, "A separable reversible data hiding in encrypted im age with improved performance", Emerging Research Areas and 2013 International Conference on Microelectronics, Communications and Renewable Energy(AICERA/ICMiCR), 2013 Annual International Conference l'IEEE 2013.
[5] Siddharth Malik, Anjali Sardana, Jaya, "A Keyless Approach to Image Encryption", 2012 international conference on Communication systems and Network Technologies l'2012 IEEE
[6] Yi-Hui Chen, Ci-Wei lan and Chiao Chih Huang, " A verifiable Visual Cryptography Scheme", Fifth International Conference and Evolutionary Computing l' IEEE 2011

DOI10.17148/IJARCCE.2017.6576