HIDING UNDELETABLE DATA IN IMAGES THROUGH CONTRAST ENHANCEMENT

#1RANGAM ANUSHA, ***Asst. Professor,***

#2PABBATHI VISHNU, ***Asst. Professor,***

Department of Electronics Communication Engineering,

Sree Chaitanya Institute Of Technological Sciences, Karimnagar, Ts.

ABSTRACT This paper describes reversible data hiding (RDH), a unique digital image approach. The proposed strategy does not place a premium on maintaining a high PSNR value. On the contrary, it improves the visual look of a primary image by increasing the disparity between its elements. The selection of the two largest bins in the histogram is done to facilitate data embedding, allowing the operation to be repeated to accomplish histogram equalization. The supplemental data is preserved alongside the message bits in the host image. This allows for the full restoration of the original image. Two sets of photographs are used to demonstrate the effectiveness of the suggested algorithm. To the best of our knowledge, this approach is the first attempt to improve picture contrast using reversible data hiding (RDH) techniques. Experiments also showed that visual integrity is preserved even when a large number of message bits are placed in images to increase their prominence. This outperforms the effectiveness of three different MATLAB methods typically used to improve image contrast.

Keywords: RDH, MATLAB, Image, Histogram.

1.INTRODUCTION

RDH has been extensively researched by signal processing experts. RDH, also known as invertible or lossless data concealment, embeds data into a host signal to create a marked signal from which the original signal can be recovered when the embedded data is extracted. RDH is beneficial in delicate situations where the host signal cannot be permanently affected. The majority of algorithms in the literature are for invisible data or digital photograph watermarks.

The hiding rate and marked image quality are used to assess the efficiency of the RDH method. There is a trade-off because increasing the hiding rate frequently alters visual content. The peak PSNR of image histogram change reduces distortion embedding capacity. Recent methods make use of neighboring pixel correlations to regulate centrally distributed prediction errors and prevent data hiding distortion. A tagged image generated with a prediction error-based approach has a high PSNR but low visual quality due to embedding distortion. High PSNR is less critical than improved image quality from low illumination. Medical or satellite images require contrast augmentation for detailed inspection. Although the psnr of the enhanced image is frequently low, image details are more visible. We are not aware of any RDH method that improves host image contrast. Rather than simply maintaining a high PSNR, the goal of this research is to develop a contrast-enhancing RDH method.

Image contrast may be improved by histogram equalization. The method modifies the pixel value histogram to include data and increase contrast. Determine the two peaks (highest segments) in the histogram. The bins on the sides are stretched outward to divide each peak into two neighboring bins, while the bins between the peaks remain untouched. To improve embedding capacity, the two highest bins in the new histogram can be divided further, and so on until a suitable contrast enhancement effect is reached. Excessive and underflowing histograms are avoided by pre-processing bounding pixel values and producing a location map to capture their places. The location map, message bits, and other auxiliary information used to reconstruct the original image are stored in the host image. As a result, raw data extraction and full image recovery are possible. Two sets of images were utilized to demonstrate the method's effectiveness. It is, as far as we are aware, the first RDH-based picture contrast enhancement algorithm. The assessment findings reveal that when many message bits are put into contrast-enhanced images, visual quality can be maintained better than three MATLAB approaches for image contrast enhancement

2.OBJECTIVE

- The primary goal of "reversible image data hiding with contrast enhancement" is to achieve true reversibility, discrete data extraction, and a significant improvement in marked image quality.
- To ensure Internet access.
- To securely transfer data, RDH employs a third-party unreadable algorithm.

3.SYSTEM DEVELOPMENNT

The payload is embedded into a digital image using reversible, lossless data embedding. Data embedding should be avoided because it can degrade image quality. Reversible data embedding improves image quality after data extraction and makes it more appealing. Reversible data embedding conceals information in digital images so that it can be recovered by authorized individuals. These measures can be used to evaluate the presentation of a reversible data-embedding algorithm:

- Data embedding capacity limit
- Visual quality
- Complexity

Because it embeds data without distortion, reversible data embedding is desirable. Data included in material will cause it to change. Even minor pixel value fluctuations might be troublesome in military and medical data. In this case, every piece of information is critical. Because the change between the implanted image and the original image is virtually undetectable, reversible data implanting can be used as a top-secret communication route. There are two types of general data hiding methods:

Method of reversible data hiding

Method for concealing irreversible data

Method for concealing data that is reversible: The original cover and message signal are recovered without loss.This approach recovers the message signal without sacrificing concealment. As a result, reversible data concealment is now available.

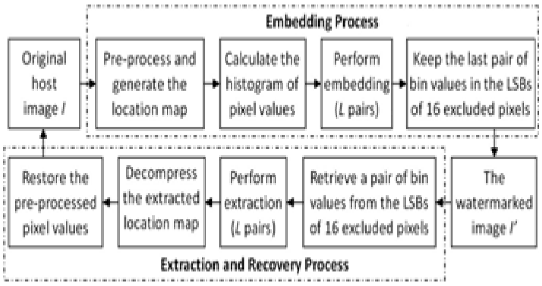

Fig. 1. The suggested RDH algorithm

Matlab's pixel value matrix process for any image is:

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. Imagine yourself as a matrix.

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

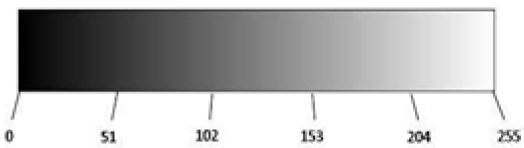

Fig 3. The grayscale scale

4.DATA EMBEDDING BY HISTOGRAM MODIFICATION

The presented algorithm is for grayscale photos, but it can simply be extended for color images. The image histogram of an 8-bit grayscale image I can be calculated by counting pixels with grayscale values j, where j € j=0,1,....254,255. hI(j) is the number of pixels with j values used to display the image's histogram.

j. Assume I have N pixel values. The two maxima (the two highest bins) are then chosen from the N nonempty bins inhI, and IS and IR denote the smaller and larger values, respectively. For count i pixels, data embedding happens.

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

Histograms display the frequency of image intensity. It displays the number of pixels for each intensity.

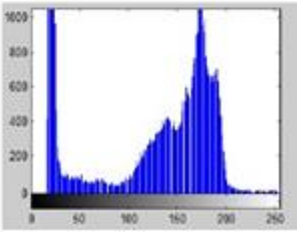

Fig 4.Original image Fig 5. Histogram

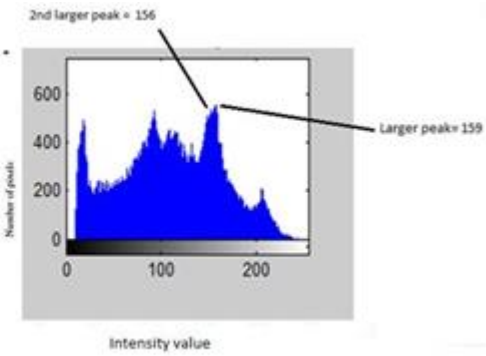

Fig 6. Histogram

A higher peak is indicated by IR. activate IR. =159 and Is =156 represent the second highest peak.

where i is the pixel value modified and bk is the kth hidden message bit (0 or 1). Using Eq. (1), each hI pixel contains binary values hI(IS) and hI(IR). The modified histogram will include N+2 segments because I has no boundary value (0 or 255). Preprocessing is required otherwise. As a result, the bins between the two peaks remain identical, while the bins on the outer are pushed outward to divide each peak into two neighboring bins (IS-1 and IS, IR and IR+1). The Matlab image scanning procedure is as follows:

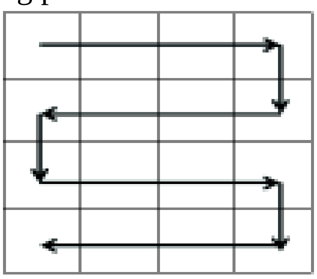

Fig 7. Scan in reverse order

Peak Is and IR are the data to embed. To retrieve embedded data, the number of Is and IR pairings is required. Boost embedding capacity. Look up "image security"

Peak values must be provided in order to retrieve embedded data. Exclude 16 pixels from the histogram calculation to keep them. The buried binary values include the least significant bits (LSB) of those pixels. After using Eq. (1) For each data embedding pixel, a bitwise operation substitutes the LSBs of the 16 missing pixels with IS and IR (8 bits). Retrieve the peak values and histogram of the indicated image after eliminating the 16 pixels to extract the embedded data.

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

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

After that, each pixel in the histogram with the values IS-1, IS, IR, and IR+1 is processed.

bK is the k-th extracted binary value from image I. The extraction and embedding processes are carried out in the same order. According to Equation. (1) To recover the original value of each histogram pixel, use the following technique:

The retrieved binary values are used to determine the original LSBs of the 16 missing pixels. To rebuild the original image, write back the deleted pixels.

Pre-Process for Complete Recovery

The method requires that all hI pixels be within a range of 1,......, 254. Any value for the enclosing pixel (0 or 255) causes the histogram to overflow or underflow. To avoid this, the histogram must be pre-processed before being modified. 0 and 255, for example, become 1 and 254, respectively. Because the possible change for each pixel is +1 or -1, no overflow or underflow will occur. A location map of the same size as the original image is built to recall the pre-processed pixels by assigning a value of 1 to a modified pixel and 0 to an unaltered pixel (including the 16 missing pixels). In disguised binary values, precomputed location maps can be used. The extracted data from the tagged image can identify modified pixels during the extraction and recovery process. By restoring the values of these pixels, the entire image may be recreated.

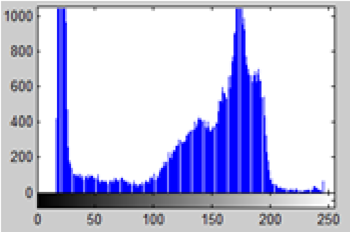

Fig 8. Histogram that has been rearranged

The main advantage of histogram shifting is that it eliminates underflow and surplus. When putting a watermark to an image, prevent negative or gray level underflow and overflow. The current system is inefficient, causing overflow and underflow, and causing distortions.

Contrast Enhancement

Contrast Enhancement modifies image pixel intensity to make the most of available bins. To improve visual contrast, the histogram of pixel values can be equalized. Determine the two highest histogram bins, ignore the bins between the peak values, then shift the outer bins outwards to separate the two peaks into two neighboring bins. This method allows for data embedding and contrast improvement. Further subdividing the top two bins of the histogram improves embedding capacity and contrast enhancement. To recreate the original image, the position map, message bits, and other data are encoded in the cover image.

Histogram equalization enhances contrast by adjusting image intensities as we add or subtract data from Is and IR. Additional peaks, Is 1 and IR.+1, are discovered. As a result, data embedding and contrast augmentation occur concurrently.

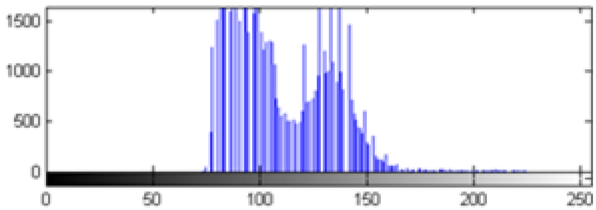

Fig 9. The first histogram

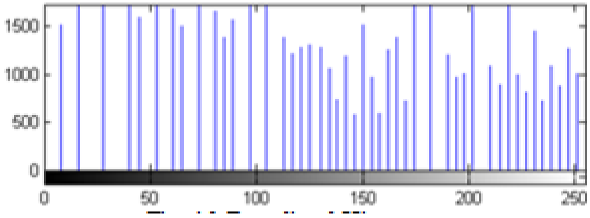

Fig. 10 Histogram with Equivalents

Because the message bits must contain about equal numbers of 0s and 1s, each of the histogram's two peaks is divided into two surrounding groups with similar or comparable heights. To boost hiding rate, the equation is employed to separate the two highest bins of the modified histogram. (1) Each pixel in the histogram. To equalize the histogram, repeat the process by splitting each peak into two surrounding bins with identical heights. As a result, data embedding and contrast augmentation occur concurrently. The pre-process adds L from 0 to L-1 to the range of pixel values and subtracts L from 256-L to 255, where L is a positive integer. By assigning 1s to modified pixels and 0s to others, a location map is constructed.

Before embedding the position map in the host image, it can be precomputed and compressed. L, the compressed location map size, the prior peak values, and the final two peaks to be divided are all contained in the LSBs of the 16 missing pixels. Eq. (2) is used to extract the final split peak values and associated data during extraction. Equation reconstructs and restores the histogram. (3) Data can be extracted from previously isolated peaks using pair-by-pair processing. The retrieved data is utilized to generate the location map in order to detect the changing pixel values from the pre-process.

5.Procedure of the Proposed Algorithm

- The procedure of the algorithm is depicted in Fig. 1. Because L pairs of histogram categories must be separated for data embedding, the procedures listed below are used:
- Except for the first 16 pixels on the bottom row, the pixels in [0 to L-1] are processed as

described before. To record the positions of these pixels, a JBIG2-compressed location map is constructed. The image histogram is computed without taking into account the first 16 pixels of the bottom row.

- To each histogram pixel, Eq. divides the histogram's two peaks (highest bins) for data embedding. The two peaks of the revised histogram are then split, and so on until L pairs are split. The bit stream of the compressed location map comes before the message bits. In the last two peaks to be divided, the compressed location map length, L, the LSBs from the 16 omitted pixels, and the prior peak values are all included.
- The marked image is created by substituting the LSBs of the 16 missing pixels with the peak values of the previous segment. The following are the steps to extract data and recover the original image:
- The LSBs of the sixteen missed pixels are used to determine the last two split peaks.
- Data is extracted from the last two split peaks using Eq. (2) in such a way that L, the compressed location map length, corresponds to the original 16-bit LSBs.
- Peak values that have previously been divided and omitted pixels are known. Then, Eq is applied to all pixels save the 16 that were omitted for recovery. The extraction and recovery processes are repeated until all split peaks have been restored and their embedded data has been extracted. The compressed location map is generated by extracting binary values and then decompressing them. The decompressed map identifies pixels that have been preprocessed and changed. We subtract L from pixels with values less than 128 and add L to those with values greater than 128. To minimize ambiguity and comply with this rule, the maximum L value is 64. The image is restored by rewriting the LSBs of the 16 deleted 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