Lossless Data Hiding For Image Using Reversible Contrast Mapping (RCM)

Warkar H.A. **Jain S.N.**

Abstract—**Data hiding is still an important research topic due to the design complexities involved. Steganography is the technique of hiding confidential information within any media. Steganography is often confused with cryptography because the two are similar in the way that they both are used to protect confidential information. But because of loss in quality of original image we propose a Reversible Contrast Mapping (RCM) is a simple integer transform that applies to pairs of pixels. For some pairs of pixels, RCM is invertible, even if the least significant bits (LSBs) of the transformed pixels are lost. The data space occupied by the LSBs is suitable for data hiding. The embedded information bit-rates of the proposed spatial domain reversible watermarking scheme are close to the highest bit-rates reported so far. The scheme does not need additional data compression, and, in terms of mathematical complexity, it appears to be the lowest complexity one proposed up to now. A very fast lookup table implementation is proposed. Robustness against cropping can be ensured as well.**

I. INTRODUCTION

MOST of the reversible watermarking approaches proposed so far incorporate a lossless data compression stage. The use of an elaborate data compression stage increases the mathematical complexity of the watermarking. There are some watermarking schemes that do not rely on additional data compression, as for instance, the circular histogram interpretation schemes, but they have the drawback of a low embedding capacity. In this letter, we discuss a spatial domain reversible watermarking scheme that achieves high-capacity data embedding without any additional data compression stage. The scheme is based on the reversible contrast mapping (RCM), a simple integer transform defined on pairs of pixels. RCM is perfectly invertible, even if the least significant bits (LSBs) of the transformed pixels are lost. The data space occupied by the LSBs is suitable for data hiding.

The basic RCM watermarking scheme was introduced in. Here, a modified version that allows robustness against cropping is proposed. The control of distortions introduced by the watermarking is investigated as well. The mathematical complexity of the RCM watermarking is further analyzed, and a very low cost implementation is proposed. Finally, the RCM scheme is compared with Tian's difference expansion scheme with respect to the bit rate hiding capacity and to the mathematical complexity. It is shown that the RCM scheme provides almost similar embedding bit-rates when compared to the difference expansion approach, but it has a considerably lower mathematical complexity.

II.RCM TRANSFORM

Let [0, L] be image gray level range (L=255 for eight-bit gray level images), and let (a, b) be a pair of pixels. The forward RCM transforms pairs of pixels into pairs of pixels

FORWORD TRANSFORM

Here below are two variables used for forword transform

$a'=2a-b,\ b'=2b-a$ (1)

To prevent overflow and Under flow, the transform is restricted to below levels

$0<= 2a-b<=L$

$0<= 2b-a<=L$

INVERSE TRANSFORM

Inverse transform is define as follows

$a= [2/3\ (a') + 1/3(b')]$ (3)

$b= [1/3\ (a') + 2/3(b')]$ (2)

As stated in Section I, the pair forward-inverse transform should give exact results, even if the LSBs of the transformed pixels are lost. If a' and b' are not changed, exactly inverts, even without the ceil functions. By watermarking, the LSBs of a', b' are lost. Let us set to "0" the LSBs of a' and b'. It immediately appears that if the LSB of a' was "1," the values inside the ceil functions for the computation of a and b decrease with 2/3 and 1/3, respectively. Similarly, if the LSB of b' was "1," the corresponding values decrease with 1/3 (for the computation of a) and 2/3 (for the computation of b). Except when both LSBs are "1," the ceil function recovers the correct results. An LSB of "1" means an odd integer number. From, it follows that (a', b') are both odd numbers only if (a, b) are odd numbers, too. To conclude, on D without the set of odd pairs, the inverse RCM transform performs exactly, even if the LSBs of the transformed pairs of pixels are lost. The forward transform should not introduce visual artifacts. By taking the sum and the difference of, one gets a'+b'=a=b and a'-b'=3(a-b), respectively. This means that RCM preserves the graylevel averages and increases the difference between the transformed pixels. Consequently, image contrast increases.

III. REVERSIBLE WATERMARKING

The watermark substitutes the LSBs of the transformed pairs. At detection, in order to extract the watermark and to restore the original pixels, each transformed pair should be correctly identified. The LSB of the first pixel of each pair is used to indicate if a pair was transformed or not: "1" for transformed pairs and "0" otherwise. The inverse RCM fails to recover the pairs (a, b) composed of odd values. Such pairs can be used as well for dataembedding as long as they are correctly identified at detection. This can be easily solved by setting the LSB of the first pixel to "0." At detection, both LSBs are set to "1" and are checked. If are fulfilled, the pair was composed of odd pixels. In order to avoid decoding ambiguities, some odd pixel pairs should be eliminated, namely, those pairs located on the borders of. The pairs subject to ambiguity are found by solving in odd numbers the equations: 2a-b=1, 2b-a=1,2a-b=L and 2b-a=L. For L=255 there are only 170 such pairs.

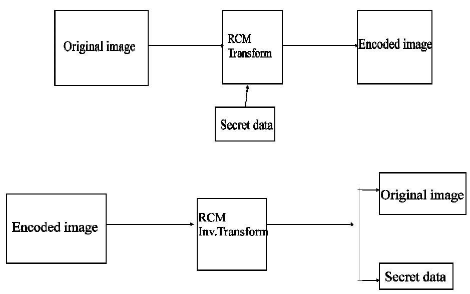


Fig. 1. Encoding and decoding.

IV. CODING

The coding proceeds as follows. Partition the entire image into pairs of pixels (for instance, on rows, on columns, or on any space filling curve).

1) For each pair (a,b):

a) (a, b) $\in$ D If and if it is not composed of odd pixel values, transform the pair using the (1), set the LSB of x 'to "1," and consider the LSB of b' as available for data embedding.

b) If (a, b) $\in$ D and if it is composed of odd pixel values, set the LSB of a to "0," and consider the LSB of b as available for data embedding.

c) If, (a, b) $\notin$ D set the LSB of a to "0," and save the true value.

2) Mark the image by simple overwriting the bits identified in 2a and 2b with the bits of the watermark (payload and bits saved in 2c).A different marking procedure is proposed in. A map of transformed pairs and the sequence of LSBs for all non-transformed pairs are first collected. Then, the entire image LSB plane is overwritten by the payload and by the collected bit sequences.

The slightly modified procedure proposed in this the embedding operation. Y-channel is utilized for data embedding. In the first step, frame selection is performed and the selected frames are processed block-wise. For each block, only a single bit is hidden. After obtaining 8 × 8 DCT of the block, energy check is performed on the coefficients that are predefined in a mask. Selected coefficients of variable length are used to hide data bit m. m is a member of message bits or frame synchronization markers.

Message sequence of each group is obtained by using RA codes for T consecutive frames. Each block is assigned to one of these groups at the beginning. After the inverse transform host frame is obtained.

Thus, all the information needed to recover any original pixel pair is embedded into the pair itself or very close to it. In the case of cropping, except for the borders where some errors may appear, the original pixels of the cropped image are exactly recovered together with the embedded payload. For pixel pairing on row or column direction, there are no problems of synchronization. Some control codes should be inserted in the payload to validate watermark integrity.

V. DETECTION AND ORIGINAL RECOVERY

Watermark extraction and exact recovery of the original image is performed as follows.

1) Partition the entire image into pairs of pixels.

2) For each pair (a', b'):

a) If the LSB of a' is "1," extract the LSB of b'and store it into the detected watermark sequence, set LSBs of a', b' to "0," and recover original pair (a, b) by inverse transform.

b) If the LSB of a' is "0" and pair(a', b') with the LSBs set to "1" belongs to D , extract the LSB of , store it into the detected watermark sequence, and restore the original pair as with the LSBs set to "1."

c) If the LSB of a' is "0" and the pair (a', b') with the LSBs set to "1" does not belong to D, the original pair is recovered by replacing the LSB of a' with the corresponding true value extracted from the Watermark sequence.

VI. STEGANOGRAPHY

The objective of steganography is to hide a secret message within a cover-media in such way that others cannot discern the presence of hidden message. Technically in simple words "steganography means hiding one piece of data within another". Modern steganography uses the opportunity of hiding information into digital multimedia files and also at the network packet level. Hiding information into a media requires following elements

1) The cover media(C) that will hold the hidden data.

2) The secret message (M) may be plain text, cipher text or any type of data.

3) The stego function (Fe) and its inverse (Fe-1).

4) An optional stego-key(K) or password may be used to hide and unhide the message. The stego function operates over cover media and the message (to be hidden) along with a stego- key (optionally) to produce stego media(s).The schematic of steganographic operation is shown below-

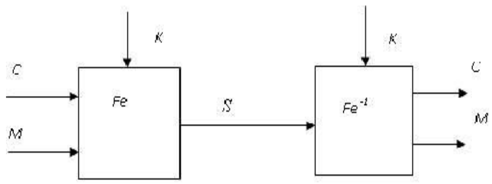


Fig. 2.steganographic operation.

The most widely technique today ishiding of secret message into a digitalimage. This technique exploits the weakness of human visual system. HVS cannot detect variation in luminance of color vectors at higher frequency side of visual spectrum. A picture can be represented by the collection of color pixels. The individual pixels can be represented by their optical characteristics like brightness, Chroma etc. Each of these characteristics can be digitally expressed in terms of 1's and 0's.

This technique can be directly applied on digital image on bitmap format as well as the compressed image format like JPEG. In JPEG format, each pixel of the image is digitally coded using discrete cosine transformation (DCT). The LSB of encoded DCT components can be used as the carriers of the hidden message.

The details of above techniques are explained below:
Modification of LSB of a cover image in 'bitmap' format.
In this method binary equivalent of the message (to be hidden) is distributed
among the LSBs of each pixel. For example we will try to hide the character 'A' into an 8-bit color image.
We are taking eight consecutive pixels from top left corner of the image. The
equivalent binary bit pattern of those pixels may be like this: -

00100111 11101001 11001000 00100111 11001000 11101001
11001000 00100111

Then each bit of binary equivalence of letter 'A' i.e. **01100101** are copied serially
(from the left hand side) to the LSB's of equivalent binary pattern of pixels,
resulting the bit pattern will become like this: -
00100110 11101001 11001001 00100110 11001000 11101001
11001000 00100111

While at the detection end extract the last bit of every pixel to get the equivalent binary pattern of hidden data.
This operation can perform by using masking operation. Mask the data with 0x1 so that we can get the last bit of each pixel. The mask operation is as follows -

00100110&00000001 = 0
11101001& 00000001 = 1
11001001& 00000001 = 1
00100110& 00000001 = 0
11001000& 00000001 = 0
11101001& 00000001 = 1
11001000& 00000001 = 0
00100111& 00000001 = 1

So this way can able to extract the hidden data 'A' i.e. **01100101.**But during this process we are not able to recover the one bit of original image or data which cause loss of quality of original image. So the problem with this technique is that it is very vulnerable to attacks such as image compressing and formatting.
Disadvantage of this technique is that the properties of electronic media are being changed after hiding any object into that. This can result in the form of degradation in terms of quality or unusual characteristics of the media: Steg analysis techniques based on unusual pattern in the media or Visual Detection of the same.

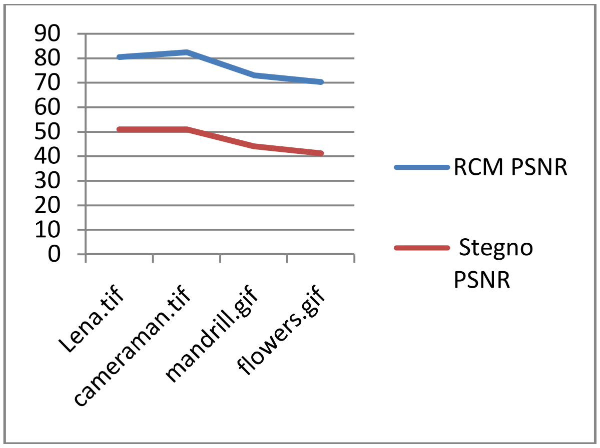
VII. Experimental Results

The PSNR block computes the peak signal-to-noise ratio, in decibels, between two images. This ratio is often used as a quality measurement between the original and a compressed image. The higher the PSNR, the better the quality of the compressed or reconstructed image.

The Mean Square Error (MSE) and the Peak Signal to Noise Ratio (PSNR) are the two error metrics used to compare image compression quality. The MSE represents the cumulative squared error between the compressed and the original image, whereas PSNR represents a measure of the peak error. The lower the value of MSE, the lower the error.To compute the PSNR, the block first calculates the mean-squared error.

Fig. 3. Test images.

Images	RCM PSNR	Stegno PSNR
Lena.tif	80.4	51.03
cameraman.tif	82.43	51.03
mandrill.gif	73.02	44.08
flowers.gif	70.23	41.25



CONCLUSION

A spatial domain reversible watermarking providing high data embedding bit-rate at a very low mathematical complexity has been discussed. As there is loss in quality of original image by steganography we propose a Reversible Contrast Mapping (RCM) is a simple integer transform that applies to pairs of pixels. The proposed scheme does not need additional data compression. In Reversible Contrast Mapping there is no loss in quality of original image. In terms of mathematical complexity, the proposed reversible watermarking appears as being the lowest complexity scheme proposed so far. The computational complexity is reduced for both coding and decoding by using LUT access for each pair of pixels and some low complexity bit manipulation. This makes our scheme very appropriate for real-time applications. Finally, by distributing the location map and by storing the saved true values close to the corresponding pixel pairs, the RCM scheme provides robustness against cropping.

REFERENCES

[1] Very Fast Watermarking by Reversible Contrast Mapping DinuColtuc and Jean-Marc Chassery, JUNE 2007.

[2] Unseen Visible Watermarking: A Novel Methodology for Auxiliary Information Delivery via Visual Contents Hun-Hsiang Huang, Shang-Chih Chuang, Yen-Lin Huang, and Ja-Ling Wu, Fellow, IEEE, JUNE 2009.

[3] Reversible Watermark Using the Difference Expansion of a Generalized Integer TransformAdnan M. Alattar, Member, IEEE, AUGUST 2004.

[4] Digital Image Watermarking Using Balanced Multiwavelets, APRIL 2006.

[5] Improved Watermark Payload Capacity Using DE On IPCM Macroblocks in H.264/AVC.

[6] Optimization-IntensiveWatermarkingTechniques for Decision Problems Jennifer L. Wong, Student Member, IEEE, Gang Qu, Member, IEEE, and MiodragPotkonjak, Member, IEEE, JANUARY 2004.

[7] Steganography and Steganalysis: Different Approaches.

[8] D. Coltuc and A. Tremeau, E. J. Delp and P. W. Wong, Eds., "Simple reversible watermarking schemes," in *Proc. SPIE: Security, Steganography,*2005, vol. 5681, pp.561–568.

AUTHOR'S PROFILE

Harshal A. Warkar
Dept.of Electronics Engg.
S.S.V.P.S College of engineering
Dhule, India[M.S.]-424002
harshalwarkar7@gmail.com

Prof.S.N.Jain
Dept.of Electronic Engg.
S.S.V.P.S College of engineering
Dhule, India [M.S.]-424002