MORPHOLOGICAL BORDER BASED SORTED PIXEL VALUE DIFFERENCE DATA HIDING TECHNIQUE

*T.SUSHMA, ##K.SYRIL EMMANUEL
*Assistant Professor, ##student
* ## E.C.E Department, P.V.P. Siddhartha Institute of Technology, Vijayawada.
Andhra Pradesh.
INDIA.

tsushmaece@gmail.com
syrilsimple@gmail.com

Abstract: **The present paper provides a new mechanism with two stages of efficient authentication based Morphological Border and Sorted Pixel Value Difference (MBSPVD) scheme. At First, we apply morphological operations on the image and shortlist few pixels. Then derive a difference value from two consecutive pixels by utilizing the pixel-value differencing technique (PVD). The hiding capacity of the two consecutive pixels depend upon the difference value. The remainder of the two consecutive pixels can be computed by using the modulus operation. Then secret data can be embedded into the two pixels by modifying their remainder. The values of the two consecutive pixels are scarcely changed after the embedding of the secret message by the proposed optimal alteration algorithm. Experimental results have also demonstrated that the proposed scheme is secure.**

Keywords: Image steganography, pixel value difference, digital watermarking, morphological operations.

1. INTRODUCTION

Mathematical morphology is a well-founded non-linear theory of image processing. Its geometry-oriented nature provides an efficient framework for analyzing object shape characteristics such as size and connectivity, which are not easily accessed by linear approaches. Morphological operations take geometrical shape of the image objects into consideration for analysis.

Mathematical morphology is theoretically founded on set theory. It contributes a wide range of operators to image processing, based on a few simple mathematical concepts. The operators are particularly useful for the analysis of binary images, boundary detection, noise removal, image enhancement and image segmentation. An image can be represented by a set of pixels. A morphological operation uses two images: the original data image to be analyzed and a structuring element (also called kernel) which is also a set of pixels constituting a specific shape such as a line, a disk or a square. A structuring element is characterized by a well-defined shape (such as line, segment or ball), size and origin. Its shape can be regarded as a parameter to a morphological operation.

The basic operation of a morphology-based approach is translation of a structuring element over the image and the erosion and/or dilation of the image content based on the shape of the structuring element. So far only few researchers used morphological principles in developing schemes for Digital Watermarking (DW). That is the reason the present study introduced morphological schemes for HCPI method of DW.

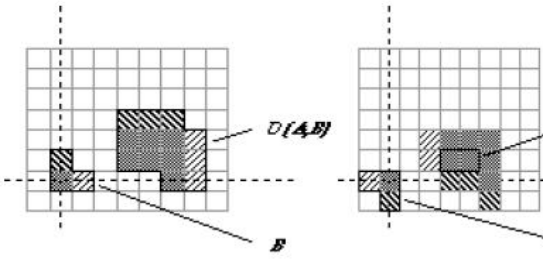
2. Fundamental Definitions

The fundamental mathematical morphology operations dilation and erosion based on Minkowski algebra are defined by the following Equations (1) and (2).

$$\text{Dilation-}\ D(A,B) = A \oplus B = \bigcup_{\beta \subset B} (A + \beta) \quad (4)$$

$$\text{Erosion}\ E(A,B) = A\Theta(-B) = \bigcup_{\beta \in B} (A - \beta) \quad (5)$$

Where these two operations are illustrated in Figure 1. While either set A or B can be thought of as an image, A is usually considered as the image and B is called a structuring element.



(a) (b)

Figure 1. A binary image containing two object sets *A* and *B*. (a) Dilation D(*A*,*B*) (b) Erosion E(*A*,*B*)

Dilation, in general, causes objects to dilate or grow in size; erosion causes objects to shrink. The amount and the way that they grow or shrink depend upon the choice of the structuring element. Dilating or eroding without specifying the structural element makes no more sense than trying to low pass filter an image without specifying the filter. Dilation and Erosion have the Commutative Non-Commutative Associative Translation Invariance and Duality properties.

Boundary extraction

Let R be a subset of pixels in an image. Then one can call R is a region of the image if R is a connected set. The boundary (also called border or contour) of a region R is the set of pixels in the region that have one or more neighbors that are not in R. If R happens to be an entire image (which is a rectangular set of pixels), then its boundary is defined as the set of pixels in the first and last rows and columns of the image. This extra definition is required because an image has no neighbors beyond its border. Normally, a region refers to a subset of an image, and any pixels in the boundary of the region that happen to coincide with the border of the image are included implicitly as part of the region boundary.

A boundary point of an object in a binary image is a point whose 4-neighborhood (or 8-neighborhood depending on the boundary classification) intersects the objects and its complement. The classification of boundaries for binary images is done by their connectivity and whether they lie within the object or its complement. Thinning methods are used for boundary transformations of the binary image. But the homotopy of the original image could not be preserved.

3. The proposed method

Instead of the difference value, the proposed scheme modifies the remainder of two consecutive pixels P(i,x) and P(i,y) for better stego-image quality. The proposed embedding and extracting algorithms are presented in the subsections below.

3.1. The embedding algorithm

Step 1: Given a sub-block Fi composed of two continuous pixels P(i,x) and P(i,y) from the cover image, obtain the difference value di, the sub-range Rj such that Rj 2 [lj,uj], the width wj = uj _ lj + 1, the hiding

capacity ti bits, and the decimal value t'i i of ti for each Fi by using Wu and Tsai's scheme according to Section 2.

Step 2: Compute the remainder values Prem(i,x), Prem(i,y) and Frem(i) of P(i,x), P(i,y) and sub-block Fi respectively

by using the following equations:

Prem(i;x) = P(i;x) mod t'i;

Prem(i;y) = P(i;y) mod t'i;

F rem(i) = (P(i;x) + P(i;y)) mod t'I;

Step 3: Embed ti bits of secret data into Fi by altering P(i,x)and P(i,y) such that F rem(i) = t'i .The optimal approach to altering the P(i,x) and P(i,y) to achieve the minimum distortion is as follows:

Case 1: F rem(i) > t'i and m ≤ (2ti)/2 and P(i,x) ≥ P(i,y)

(P'(i;x); P'(i;y)) = (P(i;x) –[m/2]; P(i;y) – [m/2]);

Case 2: F rem(i) > t'i and m ≤ (2ti)/2 and P(i,x) <P(i,y)

(P'(i;x); P'(i;y)) = (P(i;x) –[m/2]; P(i;y) – [m/2]);

Case 3: F rem(i) > t'i and m > (2ti)/2 and P(i,x) ≥ P(i,y)

(P'(i;x); P'(i;y)) = (P(i;x) +[m1/2]; P(i;y) + [m1/2]);

Case 4: F rem(i) > t'i and m>(2ti)/2 and P(i,x) <P(i,y)

(P'(i;x); P'(i;y)) = (P(i;x) +[m1/2]; P(i;y) + [m1/2]);

Case 5: F rem(i) ≤ t'i and m ≤ (2ti)/2 and P(i,x) ≥ P(i,y)

(P'(i;x); P'(i;y)) = (P(i;x) + [m/2]; P(i;y) + [m/2]);

Case 6: F rem(i) ≤ t'i and m ≤ (2ti)/2 and P(i,x) < P(i,y)

(P'(i;x); P'(i;y)) = (P(i;x) +[m/2]; P(i;y) + [m/2]);

Case 7: F rem(i) ≤ t'i and m > (2ti)/2 and P(i,x) ≥ P(i,y)

(P'(i;x); P'(i;y)) = (P(i;x) –[m1/2]; P(i;y) – [m1/2]);

Case 8: F rem(i) ≤ t'i and m > (2ti)/2 and P(i,x) < P(i,y)

(P'(i;x); P'(i;y)) = (P(i;x) –[m1/2]; P(i;y) – [m1/2]);

In the above approach, m = |F rem(i) - t'i| m1 = 2ti-|F rem(i) _ t'i| and P'(i;x); P'(i;y) are new pixel values after the embedding of ti bits of the secret data into sub-block Fi. After Step 3, if P'(i;x) or P'(i;y) overflows the boundary value ' or 255, then execute Step 4 for revising P' (i;x) and P' (i;y). If not, the purpose of concealing secret data will be completed after the replacement of (P(i,x),P(i,y)) by (P'(i;x), P'(i;y)) in the cover image.

A simple example of regulating the remainder value for hiding secret data is shown in Table 2. Suppose we have a sub-bock Fi with two successive pixel values P(i,x) = 32 and P(i,y) = 32. Then, the remainder value Frem(i) of Fi is 0. If the 3 bits (i.e. ti = 3, and t'i = 23 = 8) of the secret data are selected to be embedded into Fi, P(i,x) and P(i,y) will be modified to hold the 3-bit secret data. Table 2 demonstrates that our scheme has better performance in reducing the difference between (P(i,x), P(i,y)) and (P'(i;x); P'(i;y)). Next, we offer an example to show how our mechanism of keeping the pixel values from exceeding the range [0, 255] after secret data embedding.

As shown in Table 3, we reassume P(i,x) = 0 and P(i,y) = 0 in the previous example, and then the falling-off-boundary problem happens such that P'(i;x) < 0 or P'(i;y) < 0 when the decimal value of the secret data is 5, 6, or 7. However, P(i;x) and P'(i;y) can be re-adjusted by adding up to 4 synchronously. After that, the values of P(i;x) and P''(i;y) will fall within the range of 0–255.

4. The Extracting algorithm

In the recovery process, we can quickly extract the secret data without using the original image. It is essential to use the original range table R designed in the embedding phase in order to figure out the embedding capacity for each sub-block Fi. Given a sub-block Fi with two consecutive pixels from the stego-image with their pixel values being P(i,x) and P(i,y) respectively, the difference value di of P(i,x) and P(i,y) can be derived by Eq. (1). Each Fi can be related to its optimal sub-range Rj from the original table R according to the difference value di. Hence, we can compute the width of the sub-range by wj = uj - lj, and the number of bits ti of the secret data can be extracted from Fi by Eq. (1). Eventually, we compute the remainder value of Fi by using Eq. (2) and transform the remainder value Frem(i) into a binary string with the length ti. After that, the extracting algorithm is accomplished. For example, assume two successive pixel values of the stego-image are P(i,x) = 34 and P(i,y) = 33, and the hidden capacity is 3 bits (i.e. ti = 3). Frem(i) can be gained by (33 + 34) mod 23 = 3.

Convert the remainder value 3 into a binary string whose length is 3, and then we have 3(10) = 011(2). That is to say, the secret data 011(2) is retrieved from the sub-block in which the pixel values are P(i,x) = 34 and P(i,y) = 33.

5. EXPERIMENTAL RESULTS

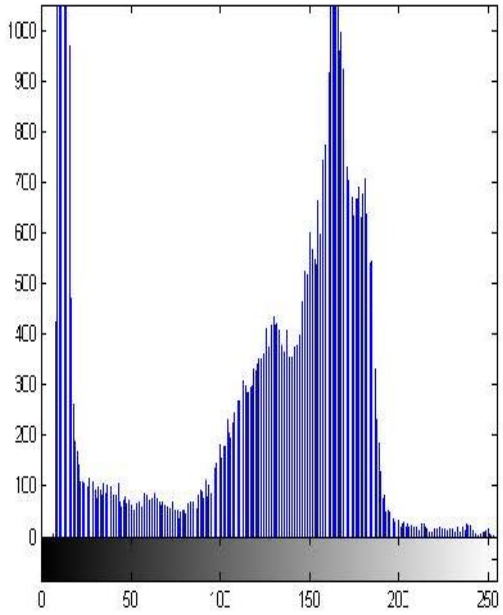
Input image

Dilated image

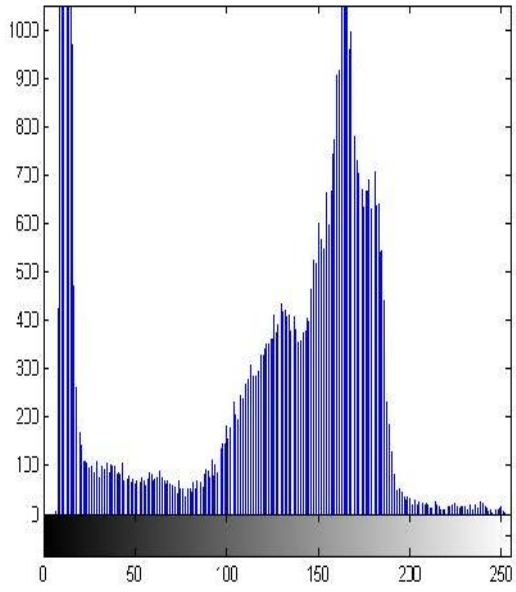
Eroded image

Morphological border image

Watermarked image



Histogram of the original image



Histogram of the watermarked image

MSE = 0.0012

PSNR = 77.29

The data is hidden into the morphological border pixels and then extracted back exactly.

6. CONCLUSION

In this paper, we proposed a novel scheme which greatly reduces the visibility of the hiding effect present in the PVD method. The proposed scheme utilizes the remainder of the two consecutive pixels to record the information of the secret data which gains more flexibility, capable of deriving the optimal remainder of the two pixels at the least distortion. The proposed method can also solve the falling-off-boundary problem by re-adjusting the remainder of the two pixels, staying secure against the RS detection attack.

REFERENCES

[1] Qiang Wu, Xiangjian He, Tom Hintz and Yuhuang Ye, A Novel and Uniform Image Separation on Spiral Architecture, International Journal of Computational Science and Engineering, Interscience, Vol.2, Nos.1/2, pp.57-63, 2006.

[2] Xiangjian He, Wenjing Jia, Namho Hur, Qiang Wu, Jinwoong Kim and Tom Hintz (2006), ―Bi-lateral

Edge Detection on a Virtual Hexagonal Structure‖, Lecture Notes in Computer Science, LNCS, Springer, Vol.4292, pp.1092-1101.

[3] http://mathworld.wolfram.com/Lagrange Interpolating Polynomial.html.

[4] Chung-Ming Wang a, Nan-I Wu a, Chwei-Shyong Tsaib, Min-Shiang Hwang (2007), ―A high quality steganographic method with pixel-value differencing and modulus function,‖ The Journal of Systems and Software.

[5] Maruthuperumal. S, Vijayakumar. V, Vijayakumar. B (2012), ―Sorted Pixel Value Difference on Fuzzy Watermarking Scheme‖, Global Journal of Science and Technology, Vol.12 Issue 4 version 1.0, February.

[6] Dong-Gyu Yeo and Hae-Yeoun Lee (2012), ―Block-based Image Authentication Algorithm Using Reversible Watermarking‖, James J. (Jong Hyuk) Park et al. (eds.), Computer Science and Convergence, Lecture Notes in Electrical Engineering 114, DOI: 10.1007/978-94-007-2792-2_69.