

A Novel Method for Image Watermarking Using Luminance Based Block Selection and Local Area Pixel Value Differencing

Sonam*, Kulbhushan Singla**

*(Department of Electronics and Communication, GTBKIET, Malout)

** (Department of Electronics and Communication, GTBKIET, Malout)

Abstract

A robust watermark scheme for copyright protection is proposed in the present paper. The present method selects the pixel locations to insert the watermark by checking luminance [1] values of blocks. The watermark is embedded in the selected pixel blocks by using local area pixel value difference method. The proposed approach overcomes the weak robustness problem of embedding the watermark in the spatial domain and also in pixel value difference method. Further the watermark extraction does not require the original image as in the case of many digital watermarking methods. The experimental results indicate the high image quality and robustness against various attacks when compared to several approaches.

Keywords: Block Selection, Digital Image Watermarking, Luminance, Pixel Value Difference, Range Table

I. Introduction

In the past few years, the need for watermarking has gained significant attention due to the spread of illegal redistribution and unauthorized use of digital multimedia. Watermarking is an information-embedding technique. Many digital watermarking schemes have been proposed for copyright protection recently due to the rapid growth of multimedia data distribution that embeds a secret imperceptible signal directly into original data, often called host signal in a robust manner. The watermark should be resistant to malicious attacks and common signal processing operations e.g. filtering, noise, and video compression. In general, the watermark contains information that uniquely identifies the content owner. Watermarking is a concept of embedding a special pattern, namely watermark, into a multimedia document so that a given piece of copyright information is permanently tied to the data.

Visible Watermark: The information is visible in the picture or video. Typically, the information is text or a logo which identifies the owner of the media.

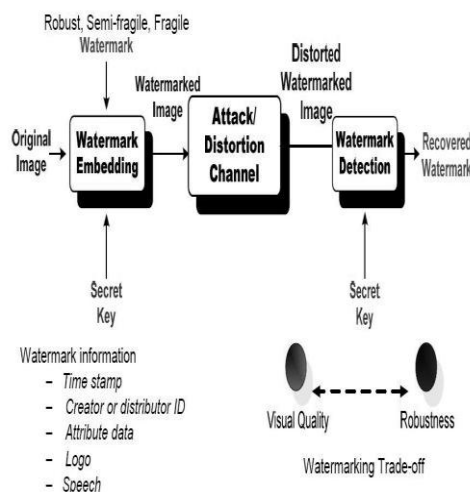
Invisible Watermark: An invisible watermark is an overlaid image which cannot be seen, but which can be detected algorithmically.

Dual Watermarking: Dual watermark is a combination of a visible and an invisible watermark. In this type of watermark, an invisible watermark is used as a backup for the visible watermark.

II. Review of PVD Method

In PVD method [2], gray scale image is used as a cover image for hiding the secret information. This

cover image is partitioned into non-overlapping blocks of two consecutive pixels, P_i and P_{i+1} . A difference value is generated by subtracting P_i from P_{i+1} in each block. The difference value is represented by 'd_i'. The set of all difference values lies a range from -255 to 255. So $|d_i|$ ranges from 0 to 255. The blocks which generates small difference values that locate in smooth area and blocks which generates large difference values that locate at the sharp edged area. The human eyes can tolerate more changes in sharp-edge area than smooth area. So, more data can be embedded into edge area than smooth area. Therefore, in PVD method, a range table has been designed with n contiguous ranges R_k . Here the value of k varies from 1 to n like $k=1,2,\dots,n$. where the range is 0 to 255.



The lower limit and the upper limit are represented by l_k and u_k respectively, then $R_k \in [l_k, u_k]$. The width of R_k is calculated using $w_k = u_k -$

l_k+1, w_k decides how many bits can be embedded into a pixel block. The embedding algorithm is given as algorithm.

- Find the difference value d_i of two consecutive pixels p_i and p_{i+1} for each segment in the cover image. This difference is given by $d_i = |p_{i+1} - p_i|$.
- Find the optimal range in which the calculated difference value lies in the range table by using d_i . This is calculated as $R_k = \min(u_k - d_i)$, where $u_k \geq d_i$ for all $1 \leq k \leq n$.
- Calculate the number of bits 't' to be embedded in a pixel segment can be defined as $t = \log_2 w_i$. Where w_i is the width of the range where the pixel difference d_i lies.
- Read t bits from binary secret data and convert it into its decimal value b.
- Now finding the new difference value d_i' using $d_i' = l_i + b$.
- Modify the values of p_i and p_{i+1} by the following method[1]:

$$(P_i', P_{i+1}') = \begin{cases} (P_i + m/2, P_{i+1} - m/2), & \text{if } P_i \geq P_{i+1} \text{ and } d_i' > d_i \\ (P_i - m/2, P_{i+1} + m/2), & \text{if } P_i < P_{i+1} \text{ and } d_i' \geq d_i \\ (P_i - m/2, P_{i+1} + m/2), & \text{if } P_i \geq P_{i+1} \text{ and } d_i' \leq d_i \end{cases}$$

$$(P_i + m/2, P_{i+1} - m/2), \\ \text{if } P_i < P_{i+1} \text{ and } d_i' \leq d_i$$

Where $m = |d_i' - d_i|$. Repeat step 1-6 until all secret data are embedded into the cover image. After embedding all secret data, a resultant image is generated which is called Stego-Image. While decoding the hidden data from the stego-image, the range table, which is used at encoding, is required. Here the same method is used for partitioning the stego-image into pixel blocks. Calculate the difference value for each block using $d_i' = |P_i' - P_{i+1}'|$. Now finding the optimum range R_i' of d_i' . Compute the b' by $b' = d_i' - l_i$. Convert b' into binary of 't' bits, where $t = \log_2 w_i$. These t bits are the hidden secret data.

III. Proposed Work

J. Hussein[1] in his work had developed a method for selecting the embedding area in cover image using Luminance of image. The image of size 512x512 is divided into 8x8 blocks and the following operations are needed and must be performed to accomplish the embedding and extracting processes:

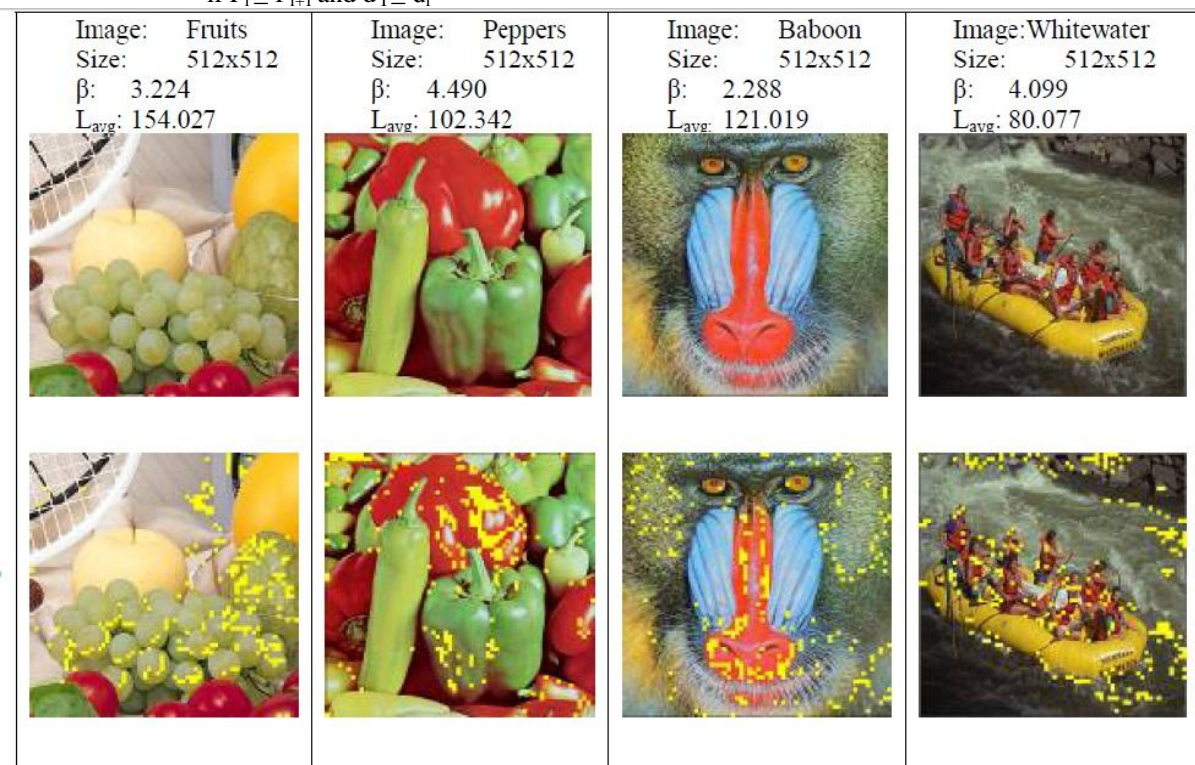


Figure 1: Cover Images and Selected Blocks (yellow)

3.1 RGB to YCbCr conversion

The RGB color space is converted to YCbCr color space for each 8x8 block using the equations (1):

$$\begin{aligned} Y &= 0.299 \times R + 0.587 \times G + 0.114 \times B \\ Cb &= 0.596 \times R - 0.275 \times G - 0.321 \times B \\ Cr &= 0.212 \times R - 0.523 \times G - 0.311 \times B \end{aligned} \quad (1)$$

where R, G, and B are red, green and blue

components of RGB color space respectively.

3.2 Log-average Luminance

The block selection criteria are depended on log-average luminance for the entire image and log-average luminance for each block. The log-average

Luminance L_{avg} is calculated as shown in the equation (2):

$$L_{avg} = \exp\left(\frac{1}{N} \log(d + Y_{x,y})\right) \quad (2)$$

L_{avg} : Log-average luminance

$Y_{x,y}$: Luminance Y of the pixel at x, y

d : A small value to avoid taking the log of a completely black pixel whose luminance is zero

N : The number of pixels in the image.

3.3. Block selection Criterion

When 512×512 host image is divided into 8×8 blocks, 4096 blocks are produced. Since the watermark size is 32×32 pixels and if we consider minimum embedding capacity of 1 bit, each block can be used to embed 36 bits minimum and thus we require only 227 blocks to embed the entire

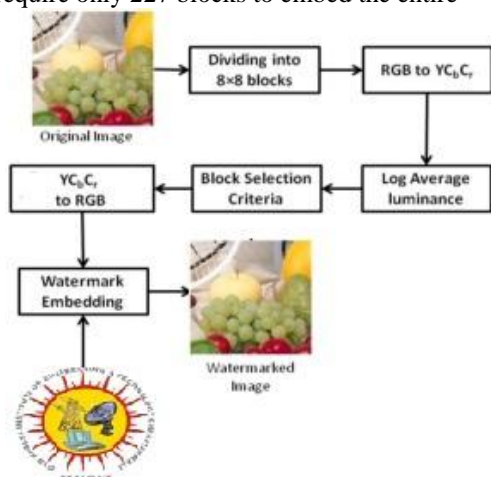


Figure 2: Watermarking Embedding

watermark. After finding the log-average luminance for the entire image and for each block; the best blocks are chosen from the blocks that have log-average luminance closer to the log-average luminance of the entire image. To do that we select the blocks with log average luminance in the range $[L_{avg}-\beta, L_{avg}+\beta,]$ where L_{avg} is the log average luminance of the image and β is the minimum floating-point value that is enough to determine adequate number of blocks.

Now in selected blocks separate RGB components of colour image. We can embed our watermark using **local area pixel value differencing**. In this method, we choose a center pixel in a matrix of 3X3 pixels (for all three colour

components). We calculate the inter pixel difference in the matrix to find out highest difference in pixel values. This highest values of difference is compared in range table and we find the maximum number of bits of information (from secret message) that we can replace in center pixel. The complete embedding and extracting method is described in following section.

3.4 Embedding Algorithm:

- Read this Cover –Image and separate its RGB components.
- Take the selected pixel in each colour domain and make a matrix of 3X3 pixel by placing selected pixel in center of matrix.
- Calculate the difference value ' d_i ' between two consecutive pixels, that are above and to the left side of center pixel.
- Now find the highest difference value among four difference values of each colour plane.
- Making the optimal range Table R_i for the d_i such that

$$R_i = \min(u_i, d_i), \text{ where } u_i \geq d_i$$

This is the optimum range where the difference lie.

- Compute the number of secret data **bits 't'** to be embedded in selected pixel pair from the width w_i of the optimum range, can be defined as $t = \log_2 w_i$.
- Read ' t ' bits from secret message and convert it into its decimal value b.
- Now finding the new difference value d_i' using $m = d_i' = l_i + b$.
- Modify the values of p_i and P_{i+1} by the original PVD method as discussed in review section. (where p_i and P_{i+1} are R and B components). Here only R and B components are selected because these two components add very little to Y value and uniformly distributing the difference does not alter their values to much extent and Y values remains almost same
- Now repeat the steps for next pixel that is immediate neighbour to center pixel and repeat the process for all pixels moving to right side and then down in the selected block of image.

3.5 Extraction Algorithm

For the extraction process, the blocks are selected in the same manner as discussed in embedding procedure. While decoding the data from the watermarked-image, the range table, which is used at encoding, is required. The following steps used for extracting the hidden data:

- Read this Cover –Image and separate its RGB components.

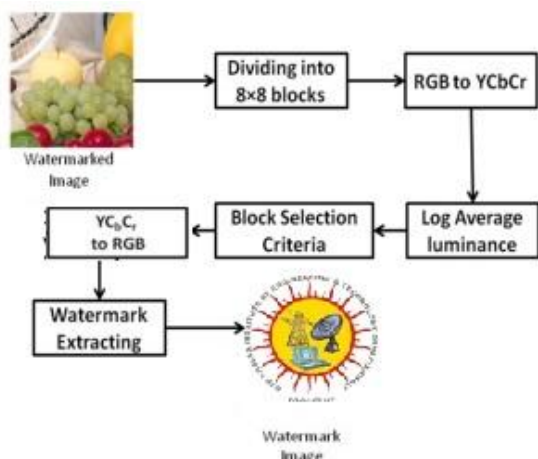


Figure 3: Extraction Process

- Starting from the last pixel, consider it as center pixel of selected block for all three colour components
- Now calculate the difference values of other pixels of matrix
- Now find the greatest difference value among the differences. Find the appropriate range R_i for the difference d_i .
- Compute the b' by $b' = d_i - l_i$. Convert b' into binary of t' bits, where $t = \log_2 w_i$. These t bits are the hidden secret data.
- Now repeat the process for other pixels by moving right to left and then to top for an image and extract the message bits before the previously extracted bits.
- After that these binary information is converted into watermark image



Figure 4 : Watermark Image

The proposed method is developed using MATLAB.

IV. Experimental Results

The proposed method is tested with various attacks and the quality parameters are listed in tables. Table 1, shows the comparison between proposed method and J.A. Hussein's [1] method. The comparison shows that the proposed method is better in performance and shows improved PSNR, MSE values. Table 2, shows the MSE, PSNR and Similarity Values with various attacks using the proposed method on the watermark images. The values of Table 2 clearly indicate the robustness and quality of the image is not degraded for all attacks. The efficiency of the proposed method in this paper is compared against luminance based

method[1]. The results, as presented in Table 1, show good improvement in similarity between the original and the extracted watermark in comparison with the previous methods.

In Table 2, four different images (Fruits, Peppers, Baboon, White water) have been used to test the robustness of the proposed watermarking method against variety of attacks. The extracted watermark image, MSE, PSNR, and Similarity are presented for each test as test parameters. These test parameters show reasonable watermark resistance against most of the attacks.

Table 1: PSNR Comparison with previous method

Images	J.A.Hussein's[1] method	Proposed Method	
	PSNR	PSNR	MSE
Fruits	32.512	50.09	0.79
Peppers	35.000	46.95	1.14
Baboon	34.990	48.08	1.00
Whitewater	32.650	48.86	0.91

V. Conclusion

In this research paper a new approach is proposed based on log-average luminance and local area pixel value differencing. This approach achieves high imperceptibility and robustness of the embedded watermark. The watermark imperceptibility is obtained by using log-average luminance [1]. Only the blocks with log-average luminance close to the log-average luminance of the entire image are used to embed the watermark. The test results show that these blocks do not degrade the image when the pixels' luminance value is increased or decreased. Also, these blocks are less affected when various filters are applied to the image.

References

- [1] J.A.Hussein, "Luminance-based Embedding Approach for Color Image Watermarking", *I.J. Image, Graphics and Signal Processing*, 2012, 3, 49-55
- [2] J. K. Mandal and Debashis Das, "A Novel Invisible Watermarking Based on Cascaded PVD Integrated LSB Technique." *springer Volume 305*, 2012, pp 262-268
- [3] J. K. Mandal and Debashis Das, "Steganography Using Adaptive Pixel Value Differencing (APVD) of Gray Images Through Exclusion of Overflow/Underflow." *CCSEA, SEA, CLOUD, DKMP, CS & IT 05*, pp. 93-102, 2012.
- [4] K. Thongkor and T. Amornraksa, "Method of Image Watermarking by Dividing Watermark into Two Chrominance Channels", *IEEE ISCIT conference 2010*, pp. 498-503.
- [5] J. A. Hussein, "Spatial Domain

- Watermarking Scheme for Colored Images Based on Log-average Luminance”, Journal of Computing, vol. 2, No.1, 2010, pp. 100-103.
- [6] G.Thirugnanam, S. Arulselvi, —Robust Digital Image Watermarking Scheme Based on DWT and ICA| Global Journal of Computer Science and Technology, Vol.10 Issue 11(Ver.1.0) October 2010.
- [7] H. Elazhary and S.Morkos, “Blind Robust Transparent DCT-Based Digital Image Watermarking for Copyright Protection”, (IJCSIS) International Journal of Computer Science and Information Security, Vol.2010, 8, No. 7, pp.183-188.
- [8] J. Manoharan, C.Vijila and A.Sathesh, “Performance Analysis of Spatial and Frequency Domain Multiple Data Embedding Techniques towards Geometric Attacks”, International Journal of Security (IJS), Vol.4, No. 3, 2010, pp. 28-37.
- [9] K.C. Chang, C.P. Chang, P. S. Huang and T.M. Tu, “A Novel Image Steganographic Method Using Tri-way Pixel-Value Differencing,” Journal of Multimedia, Volume. 3, No. 2, June 2008.
- [10] Chung-Ming Wang a, Nan-I Wu a, Chwei-Shyong Tsai b, Min Shiang Hwang, A high quality stegano graphic method with pixel-value differencing and modulus function, The Journal of Systems and Software (2007)
- [11] P. Mohanty, P. Guturu, E. Kougianos, and N. Pati, “A Novel Invisible Color Image Watermarking Scheme using Image Adaptive Watermark Creation and Robust Insertion-Extraction”. In proceeding of the8th IEEE International Symposium on Multimedia (ISM), 2006, pp. 153-160.

Table 2 : Performance with various attacks

Images		Type of Attack							
		Highpass	Lowpass	Gaussian 4	Gaussian 12	Median 3	Median 5	Impulse 0.5	Jpeg .75
Fruits	MSE	42.9	1229.53	25.9	76.9	23.3	60.95	1026.54	25.9
	PSNR	31.9	17.5	34.10	29.5	34.85	31.2	18.10	34.10
	SIMILARITY (%)	.95	.92	.97	.81	.93	.79	.70	.99
Peppers	MSE	58.6	1963.1	41.2	77.3	40.2	56.8	1027.9	69.3
	PSNR	30.5	15.6	32.1	29.6	32	31	18	29.9
	SIMILARITY (%)	.95	.92	.97	.81	.93	.79	.70	.99
Baboon	MSE	335.2	5155.4	163.1	388.2	310.2	531.8	953.8	155.9
	PSNR	23	11	26	22.5	23.6	21	18.5	26.6
	SIMILARITY (%)	.95	.92	.97	.81	.93	.79	.70	.99