Neural Network using RGB Gray Scale Watermarking and Subsequent Union of RGB Planes Embedding in a Digital Image

Dr Neha Gupta
Assistant Professor
Symbiosis University of Applied Sciences, Indore

Article History: Received: 28 March 2019; Accepted: 20 September 2019; Published online: 31 December 2019

Abstract— Digital Watermarking with PSO and Fuzzy Logic is an attempt to find suitable location using PSO and Fuzzy Logic by looking at the surrounding pixels and adaptively adjusting the pixel intensity values to encode the watermark bits. The result obtained in this technique indicate that following adaptive insertion on the pixels after finding location by PSO is even more effective to obtain better fidelity and robustness. The inverse tradeoff between robustness and fidelity is also demising. The method proposed includes color watermarking as an extension of gray scale watermarking by dividing color image into RGB gray scale and then uniting them together.

Index Terms*—Digital Watermarking, Fuzzy Logic, PSO, robustness, fidelity, digital security.*

I. INTRODUCTION

The watermarking scheme presented in this paper consist of three main parts, image transformation into RGB gray scale planes, watermark embedding, and watermark extraction.[2] The cover image is divided into 8×8 blocks of pixels in R,G,B Planes and discrete cosine transformation (DCT) is applied to these locations. After DCT transformation, the watermark is embedded and extracted as shown in the algorithm section. The watermark in the color image can be done by splitting the color image into Red, Green an blue Gray scale versions where watermark is embedded using the same generalized formula. Later, the three planes are united to give color watermark. The scheme for Binary Scale Watermarking was presented in Inge. Cox[12] and has been revised successfully to include gray scale and colored watermarks.

II. APPROACH FOR WATERMARK EMBEDDING

Watermark Embedding (Binary and Gray scale)

(1) The cover image is divided into RGB planes with 8×8 blocks of pixels and DCT is applied to each block in each plane.
(2) The blocks in each Plane are selected for inserting the watermark are selected.
(3) A BPN is selected with one input layer containing nine neurons, one hidden layer with three neurons and one output layer containing one neuron and weights are randomly assigned. The BPN is trained by supplying nine DCT coefficients from the 8×8 block (AC1 to AC9) for each RGB plane as shown in Figure 3.1 (a) and one pre decided output DCT coefficient AC12' at the output layer neuron whose value is chosen between 10 and 20.
(4) The resultant output DCT coefficient selected for training is modified by a rule as specified in the algorithm given in section 2.3 and the 12^{th} DCT coefficient of the image block (AC12')

shown in Figure 2.1 (a) is replaced by this value in each RGB plane. This modification results in encoding a watermark by splitting its portions across RGB planes.[6]

2.2.2 Watermark extraction (Color watermark after Reunion of RGB planes)

(1) The watermarked image is divided into 8×8 blocks of pixels in each of the RGB planes & DCT of each block is obtained in each plane.. Now, the block for watermark insertion is selected.

(2) First nine DCT coefficients (AC1-AC9) are provided as inputs to the already trained BPN network at the input layer for each plane, and the output is obtained.

(3) This output is compared with the 12^{th} DCT coefficient (AC12) of the DCT converted block of the watermarked image in each plane consecutively.[8]

(4) The comparison value obtained in step 3 is used to extract the watermark portion in each of the RGB Plane,encoded during the embedding stage as shown in the algorithm section given in section 2.3.2.

DC	AC1	AC5	AC6	AC14			
AC2	AC4	AC7	AC13				
AC3	AC8	AC12					
AC9	AC11						
AC10							

Figure 2.1 (a) (8×8 block of the Cover Image)

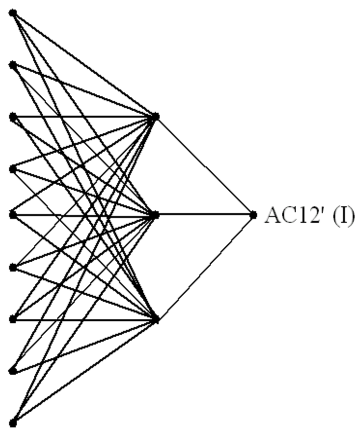

Figure 2.1 (b) BPN for encoding watermark

III ALGORITHM

3.3.1 Watermark embedding

First of all, the cover image is divided into 8×8 blocks for each RGB plane, a block is selected for watermarking and DCT is applied to this block.[10]

Let W(i) represents an array containing n values where 1<=i<=n. (This array shall be used to store n possible values of the n valued watermark).

AC12' is the selected output used for training of the backpropagation network. This process is repeated for each RGB plane separately.
(1) First the replacement value of the twelth coefficient of the cover image AC12" is obtained.

AC12" = AC12' + i× x if i th value of the watermark W(i) is to be inserted,Where 1<=i<=n & 0<X<1
(3.1)x is used to control the embedding strength of the embedded watermark).

(2) Now, the original AC12 is replaced by AC12'' as calculated in the above equation.AC12 = AC12" (3.2)
This provides encoding of RGB components of the color watermark in the three planes.
Gray Scale Watermarking for Each RGB Plane: For n=256, this technique provides a gray scale watermarking scheme, in which a gray scale watermark with intensity values from 0 to 255 may be embedded into the cover image DCT coefficients. W(1)… W(256) represents values from 0 to 255.[9] The three gray scales of RGB correspond to this scheme.

After embedding the watermark, Inverse digital cosine transform (IDCT) of the 8×8 block is taken to convert it back into the spatial domain for each plane and the three RGB planes are united to give the color watermark.

3.3.2 Watermark Extraction

First of all the watermarked image is converted into 8×8 blocks in all the three RGB planes, DCT is applied to all the blocks and the block containing the watermark is selected. Now, the following steps are taken.[10]
(1)The nine coefficients (AC1 to AC9) are
provided as inputs to the trained BPN network and the output AC12' is obtained from the trained network. This process is repeated for the three RGB planes.
(2) The inserted watermark is obtained using the following equation.
W(i) = W(k) if AC12-AC12' = k×x , for 1<=i<=n (3.3)
Where, AC12 is the 12^{th} DCT coefficient of the selected block and W(k) is the k_{th} value of watermark inserted into this block from the array W(i).
Thus, n different values of watermark which are described by an array can be embedded and successfully extracted from a given cover image. For n=256, a gray scale watermark can be used in each of the RGB planes. The three gray scale sections of the watermark in R,G,B planes can be joined together to constitute the color watermark.

IV EXPERIMENTS AND RESULT

4.1 Experimental Setup:

All the experiments were conducted on genuine intel (R) CPU T-2050 @1.60GHZ, 504 MB of RAM. The operating system used was Microsoft Windows XP Home edition, Version 2002, Service Pack 2. For conducting the experiment, a backpropagation network (BPN) was selected having one input layer, one hidden layer and one output layer.
The input layer contained 9 neurons, hidden layer contained 3 neurons and output layer contained 1 neuron. The learning rate was selected as 4. The cover image was Lena of size

(127×128) pixels and the binary and gray scale watermark was logo's image of size (127×128) pixels respectively.
The backpropagation network was trained as per the standard procedure given in the Appendix B and also in [13].The same logo's image was converted into a binary watermark image by thresholding operation on the gray scale logo's image with 0.5 level.

4.2. Generation of watermark

The algorithms given in sections 3.3.1 and 3.3.2 were used for embedding the watermark into the cover image to form the watermarked image and then extracting it from the watermarked image respectively**.** As the watermark is not obtained directly as a result of training, but the trained network helps to find the adjustment value of a DCT parameter of the cover image which provides the required encoding for the watermark to be inserted, the variation of PSNR of extracted watermark with threshold value used in training, training time or number of epochs used in BPN training are not recorded in this scheme.[4] The gray scale watermark thus obtained in the three R,G,B planes are united together to yield color watermark.

4.3 Experimental Results (Fidelity, Robustness and Payload)

Tables 4.1 and Figures 4.2 to 4.7 indicate the experimental results containing fidelity, robustness and payload of gray scale watermarking in each of the RGB planes gray versions. The Gray version results are almost similar in the three RGB planes. The results shown below indicate the gray scale of the Green section selected as a representative sample of the scheme.

)

Attacks:

Blurring: (3×3 averaging filter,0.5%) , Crop: (Top Left (30%), Contrast Enhanced (3×3 Contrast
enhancement filter,40%) ,JPEG Compression(CR=10.75 with QF=50%), Rotation(15degrees),Scaling(50%->(1-1/2-1),1-3-1),Gaussian noise (25%,Variance=0.1), Sharpening (30%,3×3 Laplacian filter).

	
Lena	Logo

Figure 4.5 Un attacked watermarked image (Lena) and gray scale watermark (Logo)

Table 4.1

(EMBEDDING GRAY SCALE WATERMARK IN A DIGITAL IMAGE BY ADJUSTING DCT COEFFICIENTS THROUGH BACKPROPAGATION NEURAL NETWORK

Cover Image	Watermark Image	PSNR (dB) of watermarked image(Fidelity) and NC of watermark extracted (no attack situation)	Size of watermark	Attack	PSNR of extracted watermark & NC of extracted watermark after attack
 Lena's Image	 logo's Image (Gray scale)	41.62,0.999	(127× 128) pixels with 256 gray values	Blurred (0.5%)	34.4,0.899
				3×3 Averaging filter	34.29,0.897
				Cropped (30%)	42.3,0.982
				Sharpened (30%)	39.6,.912
				3×3 Laplacian filter	39.78,0.952
				Compressed (CR=10.75 QF=50%)	31.7,0.861
				Gaussian noise-25%	31.59,0.860
				Variance=0.1	30.22,0.854
				Contrast Enhanced (40%)	33.21,0.897
				3×3 contrast enhancement filter	33.94,0.899
				Rotated 15^0	31.23,0.855
				Scaled (50%) 1-½-1	26.9,0.768
				1-3-1	27.8,0.779

				
Blur (0.5%)	blur (3×3 averaging filter)	Sharpened (30%)	sharpened (3×3 laplacian)	Contrast enhancement 40%
				
Contrast enhancement filter (3×3)	scaling 50% 1-1/2-1	Scaling 1-3-1	gaussian 25%	gaussian variance=0.1
				
cropped 30%	rotate 15^0	Compressed CR=10.75,QF=50%		

Figure 4.6 Attacked watermarked images (Lena)

				
blur 0.5%	**blur 3×3 averaging filter**	**Sharpened 30%**	**Sharpened 3×3 laplacian**	**Contrast enhanced 40%**
				
Contrast enhancement filter 3×3	**Scale 50% (1-1/2-1)**	**Scale 1-3-1**	**Gaussian 25%**	**Gaussian Variance=0.1**
				
Crop 30%	**Rotate 15^0**	**Compressed (CR=10.75, QF = 50%)**		

Figure 4.7 Watermark images extracted after various attacks (Logo)

4.4.4 Trade off (Robustness and Fidelity)

The fidelity of the watermarked image is affected by changing the embedding strength of the watermark by varying the value of x used in eq. (3.1) of section (3.3.1) from 0.1 to 0.3. The results are shown in the Tables 3.3 and 3.4 for the binary watermarking and the gray scale watermarking schemes respectively. The results shown in Tables 3.1 and 3.2 are shown for the strongest insertion of watermark corresponding to x=0.3.

Table 4.4 Trade off – Robustness and Fidelity (Gray scale watermarking)

S-No	Fidelity Watermarked Image-PSNR(dB) (Unattacked)	Normalized Correlation (Extracted Watermark) (Unattacked)	Trade off
1	41.62 (x=0.3)	.999	Yes
2	42.78 (x=0.2)	.976	
3	44.91 (x=0.1)	.971	

V. DISCUSSON

The PSNR and NC of the output watermark of Logo's image varies with different kinds of attacks. Normalized correlation of extracted watermark decreases with increase in the fidelity of the image by controlling the embedding strength under no attack situation showing the existence of tradeoff between the two. The payload of watermark selected is as high as the size of the watermark image of Logo (127×128) pixels binary and (127×128) pixels with 256 gray values in the binary and gray scale watermarking schemes respectively. These correspond to ($127\times128\times2$) bits and ($127\times128\times8$) bits respectively. Thus the scheme can be used with reasonable value of tradeoff and allowable robustness and fidelity values by dividing the color image into R,G,B planes gray scale versions and inserting watermark in each section and then uniting them together to yield color watermark.

VI.REFERENCES

[1] R.Schyndel, A.Tirkel and C.Osborne, "A Digital Watermark" in Proc. IEEE International conference on Image Processing, 1994, vol.2, pp.86-92.

[2] Xia-Mu Niu and Sheng-He Sun, "Multiresolution Digital Watermarking for Still Image" in Proc. IEEE Neural Networks for Signal Processing, 2000, vol.2,

[3] Ping Dong, Jovan G.Brankov, Nilolas Palastsanos, Yongyi Yang, Franck Davoine,"Signal Compression Digital Watermarking Robust Geometric Distortions", IEEE Transaction on image processing, December 2005, vol.14(12).

[4] Charkari N.M. and Chahooki M.A.Z., "A Robust High Capacity Watermarking Based on DCT and Spread Spectrum" in IEEE International Symposium of Signal Processing and Information Technology, 2007, pp.194-197.

[5] Chu-Hsing Lin, Jung-Chun Liu, Chih-HsiongShihandYan- Wei Lee, "A Robust WatermarkScheme for Copyright Protection" in MUE International Conference on Multimedia and Ubiquitous Engineering, 200 pp 132-137.

[6] Larijani H.H.and Rad G.R., "A New Spatial Domain Algorithm for Gray Scale Images Watermarking" "ICCCE International conference on computer and communication engineering", 2008, pp. 157-161.

[7] Fredric M.Ham and Ivica Kostanic, “Principles of Neurocomputing for Science & Engineering”, Mc.GrawHill, Singapore, 2001, pp. 136-140.
[8] Yu,P.T., Tsai H.H., and Lin J.S., “Digital Watermarking based on Neural Networks for Color Images , Signal processing, vol.81, pp.663-671.
[9] J.R.Hernandez, F.Perez Gonzalez and J.M.Rodriguez, “Data Hiding for Copyright Protection of Still Images”, National conference in image processing, Faislabad,2001.
[10] Hwang M.S., Chang C.C. and Hwang K.F.,”Digital Watermarking of Images using Neural Networks”,Journal of electronic imaging, 2000, vol. 9,pp.548-555.
[11] CharrierM., Cruz,D.S. and Larsson M., “JPEG 2000 , the Next Millennium Compression Standard for Still Images” in Proc. IEEE International Conference.
[12] Inge. Cox Book on Watermarking.