ISSN:2321-1156
International Journal of Innovative Research in Technology & Science(IJIRTS)

Implementation of Digital Watermarking Using Discrete Wavelet Transform

Baisakhi Das, das_baisakhi@yahoo.com; GuruNanak Institute of Technology, Sodepur, Kolkata 700114

Abstract

Powerful digital signal processing techniques and the rapid growth of Internet have made the world shift towards digital representation of multimedia signals, such as image, audio and video, however, with a fear in the mind of the originators, of the illegal distribution and violation of copyright protection by malicious users. Digital watermarking, more specifically, the hidden digital watermarking technique, however, comes into the rescue, as a powerful solution to such potential problems. Several hidden type watermarking techniques have been proposed with a variety of their usage, complexity and security – which are the primary concerns of such technique. In this paper, we have proposed a hidden type digital watermarking architecture based on discrete wavelet transform, where it is found that data embedding in approximation sub-band offers data trade off among the different watermarking requirements. The scheme includes the LZW compressed watermark to ensure high data embedding rate along with the good quality of imperceptibility. The use of wavelet transform is sufficed for the data concealment with its indiscriminate robustness property. An algorithm has been developed and programmatically implemented and the corresponding results are shown in the result section.

Introduction

With the global and wide use of internet and different network topologies, different data and media types have become less protected, and hence, could be easily downloaded and modified, by any malicious user with his/her own accord. Because of such threats, several copyright problems appeared lately and different watermarking techniques were proposed. One of them is by adding a visible watermark to the cover image, while the other one is to use a watermark technique that will embed a hidden watermark within the cover, where the watermark is the message that will be embedded into the pixels of cover image, while the cover image is the image that will be watermarked using the watermarked image to protect it from being claimed by other person or group.

In the past, duplicating art work was quite complicated and required a high level of expertise for the counterfeit to look like the original. However, in the digital world this is not true. Now it is possible for almost anyone to duplicate or manipulate digital data and not lose data quality. Similar to the process when artists creatively signed their paintings with a brush to claim copyrights, artists of today can watermark their work by hiding their name within the image. Hence, the embedded watermark permits identification of the owner of the work. It is clear that this concept is also applicable to other media such as digital video and audio[3].

In this paper, we have implemented the digital watermarking using discrete wavelet transform. For watermarking, data has to be hidden inside the cover image. Each image is presented mathematically by matrix of numbers. First, the RGB image is converted into gray image and then the corresponding pixel value of gray image is extracted, discrete wavelet transform is applied on the cover image that is the image which is to be watermarked; as a result the image is divided into four parts of equal size. These are approximation, diagonal details, horizontal details and vertical details respectively. This is done using **Haar Wavelet Transform (HWT)** considering the first 64 pixel, 8x8 matrix. The watermarking image is then compressed using the **LZW** compression technique, which compresses the image, the ASCII value of which is extracted and converted into corresponding decimal equivalent pixel value. Then the **SVD** value of both the cover image and watermark image is extracted and then the S value of cover image is substituted with the S value of watermarked image. Now the inverse wavelet transform is applied on this approximation part along with remaining three parts, and as a result, watermarked image is obtained.[9]

I. WAVELET TRANSFORMATION

For time domain signal, the time amplitude representation is not always the best representation of the signal for most signal processing related application, which is also true for 2-D image as well. The pixel or space domain representation is not always the best representation. In many cases the most distinguished information is hidden in the frequency content of the signal. The frequency spectrum of a signal is basically the frequency components (spectral components) of that signal i.e. it shows what frequencies exists in the signal and with the help of Fourier Transform (FT) we can measure frequency or find the frequency content of a signal.[11]

A. Fourier Transform and Fast Fourier Transform

The FT is a reversible transform, i.e. it allows going back and forward between the raw and processed (transformed) signals but, only either of them is available at any given space (time). That is no frequency information available in the space (time)- domain signal, and no time information is available in the Fourier transformed signal. [1]

However, the other alternative, viz., the fast Fourier transform (FFT) is an efficient algorithm to compute the discrete Fourier transform (DFT) and it's inverse, as well. A DFT decomposes a sequence of values into components of different frequencies. This operation is useful in many fields, but computing it directly from the definition is often too slow to be practical. An FFT is a way to compute the same result more quickly: computing a DFT of N points in the obvious way, using the definition, takes $O(N^2)$ arithmetical operations, while an FFT can compute the same result in only $O(N \log N)$ operations. The difference in speed can be substantial, especially for long data sets where N may be in the thousands or millions—in practice, the computation time can be reduced by several orders of magnitude in such cases, and the improvement is roughly proportional to $N/\log(N)$. The most well known FFT algorithms depend upon the factorization of N, but (contrary to popular misconception) there are FFTs with $O(N \log N)$ complexity for all N, even for prime N. Many FFT algorithms only depend on the fact that $e^{-\frac{2\pi i}{N}}$ is an N^{th} primitive root of unity, and thus can be applied to analogous transforms over any finite field, such as number-theoretic transforms. Since the inverse DFT is the same as the DFT, but with the opposite sign in the exponent and a $1/N$ factor, any FFT algorithm can easily be adapted for it.[7]

B. Continuous Wavelet Transform (CWT)

The space (time) frequency resolution problems are results of a physical phenomenon, the Heisenberg's Uncertainty Principle, and exist regardless of the transform used. But it is possible to analyze any signal by using an approach called Multi Resolution Analysis (MRA). MRA is defined to give good space (time) resolution and poor frequency resolution at higher frequencies and poor space (time) resolution and good frequency resolution at low frequencies. This philosophy has been taken into account in the Continuous Wavelet Transform.

The Continuous Wavelet Transform is developed as an alternative approach to the STFT to overcome the resolution problem in a similar way (signal is multiplied with a function, the wavelet, similar to window function of STFT) but major differences of CWT & STFT are as follows:

Negative frequency components are absent,

Sine or Cosine waves are not taken as the basis function,

Width of window is changed as transform is calculated for every single spectral component.[12]

The continuous transform is defined as follows:

$CWT_x^{\Psi}(a,b) = \int f(t)\,(1/\sqrt{(|a|)})\Psi^*((t-b)/a)dt, \quad \infty<t<\infty$

The transformed signal is a function of two variables, a and b, the translation and scale/ dilation parameters respectively. $\Psi(t)$ is a transforming function and is called the mother wavelet.

As mentioned before the CWT maps a one-dimensional signal to a two-dimensional time-scale joint representation that is highly redundant. The time-bandwidth product of the CWT is the square of that of the signal and for most applications, which seek a signal description with as few components as possible, this is not efficient. To overcome this problem *discrete wavelets* have been introduced. Discrete wavelets are not continuously scalable and translatable but can only be scaled and translated in discrete steps. This is achieved by modifying the wavelet representation.[10]

$$\Psi_{j,k}(t) = \frac{1}{\sqrt{s_0^j}}\Psi\left(\frac{t - k\tau_0 s_0^j}{s_0^j}\right)$$

C. *Haar wavelet Algorithm*

Procedure for Haar Wavelet Transform:

1. Find the average of each pair of sample (n/2 avg)
2. Find the difference between each average and sample of it was calculated from n/2 differences)
3. Fill the first half of the array with averages.
4. Fill the second half of the array with differences.
5. Repeat the process on the first half of the array. (The array length should be power of 2)

D. The LZW Algorithm

The LZW coding assigns fixed-length code word to variable length sequence of source symbols but requires no a prior knowledge of probability of occurrence of the symbol to be encoded. LZW coding is conceptually very simple. At the onset of the coding process, a codebook or a "dictionary" containing the source symbol to be coded is constructed. For 8 bit monochromatic image, the first 256 word of the dictionary are assigned to the gray values 0,1,2,….256.As the encoder sequentially examines the image's pixel, gray-level sequence that are not in the dictionary are placed algorithmically determined locations. If the first two pixels of the images are white, for instance, sequence "255-255" might be assigned to location 256 ,the address following the location reserved for gray level 0 through 255.the next time that two consecutive while are encountered, code word 256,the address of the location containing sequence 255-255,is used to represent them. If a 9-bit, 512-word dictionary is employed

in the coding process, the original (8+8) bit that were used to represent the two pixel are replaced by single 9 –bit code word. Clearly, the size of the dictionary is an important system parameter. If it is too small, the detection of matching gray-level sequence will be less likely; if it is too large, the size of the code words will adversely affect compress performance.[8]

II. THE PROPOSED ALGORITHM

The proposed hidden type watermarking algorithm, which is based on wavelet transform, is stated below.

For Watermarking, data has to be hidden inside the cover image.

- First discrete 2-D Wavelet transformation is applied on this image. As a result the image is divided into four parts of equal size. These are approximation, diagonal details, horizontal details and vertical details.
- LZW compression method is applied on this watermark image. As a result we get the compressed image whose pixel value is extracted.
- Apply the SVD to the Watermarked image and cover image is extracted.
- Now the S value of cover image is substituted with the S value of watermarked image.
- Now the inverse wavelet transform is applied on this approximation part along with remaining three parts and as a result watermarked image is obtained.

Schematic and Explanation

In this project digital watermarking is implemented using discrete wavelet transform.

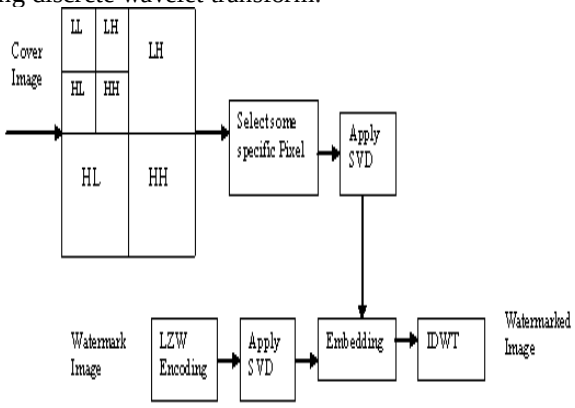

Fig.3 Watermark embedding technique

For Watermarking, data has to be hidden inside the cover image. Each image is presented mathematically by matrix of numbers .First the RGB image is converted into gray image and then the corresponding pixel value of gray image is extracted, discrete wavelet transform is applied on the cover image that is the image which is to be watermarked; as a result the image is divided into four parts of equal size. These are approximation, diagonal details, horizontal details and vertical details. This is done using Haar wavelet transform considering first 64 pixels, 8X 8 matrix. The watermarking image is then compressed using LZW compression technique, which compresses the image, the ASCII value of which is extracted and converted into corresponding decimal equivalent pixel value. Now the SVD value of both the cover image and watermark image is extracted and then the S value of cover image is substituted with the S value of watermarked image. Now the inverse wavelet transform is applied on this approximation part along with remaining three parts and as a result watermarked image is obtained.[4]

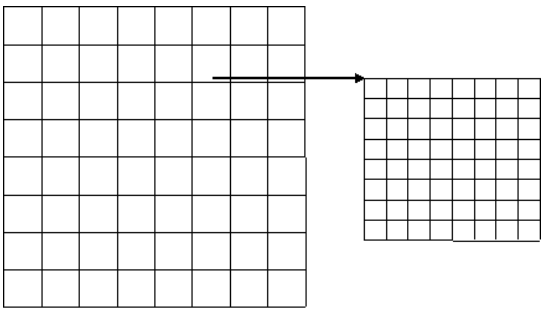

Fig.1 Depicts that some pixel value from the cover image are extracted for DWT

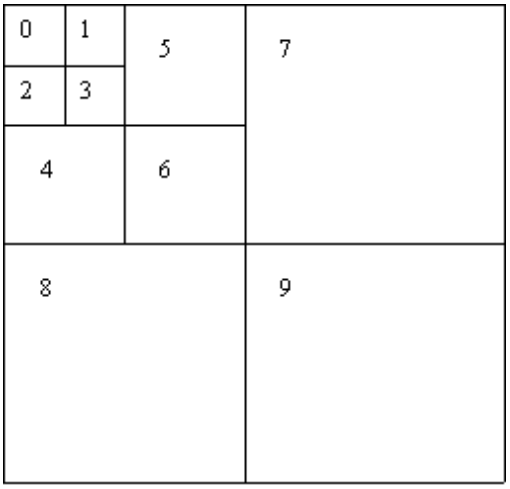

Fig.2 Depicts partition of image after Haar Wavelet Transform

III. RESULTS

Matlab Commands used:

● cd E:

The image was stored in E drive.

● A=imread('parrot.jpg','jpg');

Reading image named'parrot.jpg' from E drive.

●imshow(A);

Display the image.

●B=rgb2 gray (A);

Convert the color image to gray image

●imshow(B);

Display the image

● **B**(1:8,1:8)

Fig. 4 Original Image

We get the first 64-bit pixel value

107	116	126	127	123	117	113	109
113	118	122	120	118	116	116	115
117	116	114	111	110	112	116	119
117	111	107	106	107	109	114	117
118	111	109	114	117	115	113	114
123	117	118	129	134	125	116	114
128	123	127	141	147	134	119	113
131	124	129	145	151	137	119	112

● After Haar wavelet transform (Java Code)

119.4688	-4.5000	2.2813	-6.0313	0	1.6875	-4.0625	-1.0000
0.3750	0.1563	0.8438	-0.7813	0.3750	0.4375	-0.3125	-0.2500
-1.7188	0.5938	-4.0000	2.1875	-2.3750	-0.8750	1.6250	0.2500
4.1563	-4.5938	3.0625	-4.5000	1.8750	0.2500	-3.0000	-1.0000
1.1250	-2.0000	-2.6250	0.1250	-1.0000	-1.2500	0.2500	-0.5000
-2.5625	3.1875	-0.3750	1.7500	-0.7500	0.5000	1.5000	0.5000
2.6250	-2.1250	1.5000	-2.0000	1.0000	0	-1.7500	-0.2500
0.8125	-0.9375	1.3750	-1.5000	0.7500	0	-0.7500	-0.2500

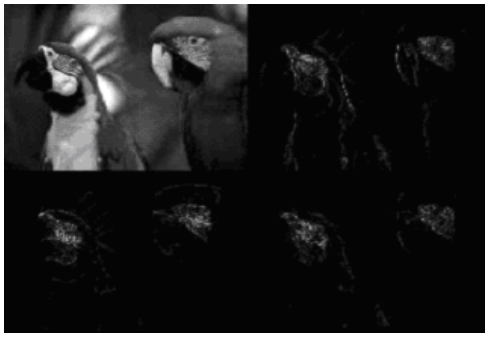

Fig. 5 Image after Haar wavelet Transform

● [U,S,V] = svd(B)

Apply SVD on the cover image, we get

U =

-0.9984	0.0532	0.0170	0.0030	-0.0010	-0.0005	0.0053	0.0023
-0.0037	-0.0945	0.1475	0.5517	-0.1159	0.3099	-0.6701	0.3262
0.0168	0.4426	-0.5222	0.4155	0.4945	-0.1832	0.1195	0.2570
-0.0396	-0.7209	-0.2186	0.4328	0.1485	0.1954	0.3688	-0.2178
-0.0093	0.0817	-0.6976	-0.0042	-0.5453	0.0046	-0.2564	-0.3788
0.0236	0.3117	0.4010	0.4929	-0.0217	-0.2446	0.0185	-0.6617
-0.0242	-0.3427	-0.0871	-0.2185	0.4906	-0.4581	-0.5643	-0.2402
-0.0082	-0.2269	0.0497	0.2119	-0.4260	-0.7500	0.1271	0.3755

S =

120.0007	0	0	0	0	0	0	0
0	10.5289	0	0	0	0	0	0
0	0	5.1644	0	0	0	0	0
0	0	0	0.7612	0	0	0	0
0	0	0	0	0.5067	0	0	0
0	0	0	0	0	0.2972	0	0
0	0	0	0	0	0	0.2186	0
0	0	0	0	0	0	0	0.0403

V =

-0.9968	0.0730	0.0156	-0.0230	-0.0172	-0.0014	0.0029	-0.0000
0.0403	0.4836	0.6685	0.2317	-0.0472	-0.2284	-0.4578	0.0054
-0.0208	-0.4838	0.6199	-0.0976	-0.0666	-0.1586	0.4241	-0.4028
0.0528	0.5268	0.0653	-0.6657	0.1218	0.3186	0.1438	-0.3677
-0.0013	-0.3102	0.2387	-0.5169	-0.4084	0.2558	-0.3205	0.4946
-0.0141	-0.0442	0.3036	0.3190	0.4397	0.7260	0.1522	0.2454
0.0357	0.3755	0.0453	0.2021	-0.6109	0.0666	0.6011	0.2756
0.0089	0.1006	0.1148	-0.2886	0.4948	-0.4735	0.3204	0.5671

For the watermark image the image is first compressed using LZW Algorithm which compresses the image, we get corresponding ASCII value as output, the ASCII value of which is converted into decimal equivalent pixel value, denoted by W. The output is as follows:

32	112	88	88	32	112	96	32;
32	32	32	70	32	32	70	32;
32	32	32	48	32	32	32	13;
10	32	32	32	32	112	97	32;
32	32	32	80	120	32	32	102;
32	32	73	32	32	42	32	32;
32	32	32	32	32	32	127	13;
20	48	32	32	49	32	32	96;
32	32	12	32	32	80	32	32;
13	46	32	32	9	32	32	13;
10	32	32	32	32	80		

Applying SVD on this pixel values of watermark image, we get, [U,S,V]=svd(W)

U =

-0.5507	-0.3480	-0.5476	-0.1144	-0.0320	0.3517	0.3137	0.1989
-0.3044	-0.0075	0.1991	-0.3693	-0.1628	0.2382	-0.7675	0.2429
-0.2228	0.0125	-0.0949	-0.3092	-0.2852	-0.0921	-0.0306	-0.8688
-0.3726	-0.3205	0.1910	0.7508	-0.2948	-0.1839	-0.1790	-0.0745
-0.3941	0.7314	0.1702	0.0101	-0.3693	-0.0555	0.3145	0.2054
-0.2661	0.0273	-0.2968	-0.1614	0.2759	-0.8288	-0.1588	0.1613
-0.3211	-0.3047	0.7015	-0.2683	0.3439	-0.0613	0.3426	-0.0216
-0.2940	0.3839	-0.0762	0.3051	0.6880	0.2890	-0.1951	-0.2690

S =

397.1226	0	0	0	0	0	0	0
0	145.1153	0	0	0	0	0	0
0	0	85.2393	0	0	0	0	0
0	0	0	67.8137	0	0	0	0
0	0	0	0	47.1530	0	0	0
0	0	0	0	0	40.1404	0	0
0	0	0	0	0	0	21.0362	0
0	0	0	0	0	0	0	10.5614

V =

-0.1901	0.0553	0.0539	-0.3715	0.0735	-0.2587	-0.2493	-0.8279
-0.3424	-0.1110	-0.4357	-0.1369	0.2901	0.5430	0.4969	-0.1893
-0.3248	-0.0881	-0.4100	-0.2659	0.3129	-0.6290	-0.0221	0.3923
-0.3830	0.1455	-0.1005	-0.4411	-0.5310	0.3400	-0.4045	0.2574
-0.3195	0.5270	0.2530	0.0158	-0.3302	-0.2724	0.6104	-0.0100
-0.4123	-0.3282	-0.2770	0.6530	-0.3850	-0.1452	-0.1108	-0.1937
-0.4753	-0.4600	0.6977	-0.0451	0.2228	0.0703	0.0145	0.1375
-0.3105	0.5989	0.0398	0.3864	0.4729	0.1635	-0.3765	0.0448

- The S value of cover image is substituted with the S value of watermarked image.

IV. Conclusion and Future Work

The proposed watermarking architecture describes an oblivious data hiding technique in the wavelet domain, where it is found that data embedding in approximation sub-band offers data trade off among the different watermarking requirements. The scheme also shows LZW compressed watermark. These ensure high data embedding rate along with the good quality of imperceptibility. The use of wavelet transform is sufficed for the data concealment with its indiscriminate robustness property.

Currently, we have performed the embedding part of watermark image in the cover image using discrete wavelet transform; however, in the next phase of our experiment, we shall try to implement the corresponding architecture using VHDL and test it on FPGA board.

References

[1] R. Anderson, Information Hiding, Proceedings of the first Workshop on Information Hiding, LNCS-1174, Springer Vela, New York, 1996.

[2] N.F. Jhonson and S. Jajodia "Exploring steganography: seeing the unseen",Computer, vol.31.pp 26-34, 1998.

[3] S. Katzenbesser and F.A.P Petitcoals, Information hiding techniques for steganography and digital watermarking, Artech House, Boston, M A ,2000.

[4] Chiou-Ting Hsu and Ja-Ling Wu, Senior Member, IEEE "Hidden Digital Watermarks in Images"-IEEE Transaction on Image Processing,Vol.8.pp58-68,1999.

[5] I .J. Cox, J. Kilian, T. Leighton, and T .Shammon, "Secure Spread Spectrum Watermarking for images, audio and video"- IEEE Pro.Int.Conf.Image Proc. 3,243.

[6] C.Hsu and J.Wu. "Hidden Signatures in Images" IEEE ICIP III'96, PP. 223-226.

[7] H . Inoue, A. Miyazaki, Ayamamoto and T.Katsura, "A digital watermark based on wavelet transform and its robustness on image compression", in Proc.ICIP'98,vol.2, Chicago, IL, Oct. 4-7, 1998 ,pp. 391-395.

[8] R . Dugad, H . Ratakonda and N . Abuja, "A new wavelet based scheme for watermarking images", in Proc. ICIP'98, vol.2, Chicago, IL, Oct. 4-7, 1998, pp. 419-423.

[9] C .I . Podilchuk and W. Zeng , "Image Adaptive Watermarking using visual models",

[10] IEEE J.Select. Areas Communication, vol.16,pp, 525-539, May 1998.

[11] Jerome M . Shapiro, "Embedded Image Coding using Wavelet coefficients",IEEE Transactions on Signal Processing. Vol . 41 No. 12, December 1993.

[12] Rao and Bopardikar, "Wavelet Transforms", Addison –Wesley.

[13] Yang Qianli , Cai Yanhong; A digital image watermarking algorithm based on discrete wavelet transform and Discrete Cosine Transform. Information Technology in Medicine and Education (ITME), 2012 International Symposium , 2012.

[14] Pathak, Y, Dehariya, S.; A more secure transmission of medical images by two label DWT and SVD based watermarking technique. Advances in Engineering and Technology Research (ICAETR), 2014