Pseudorandom Noise Sequence of Digital Watermarking Algorithm based on Discrete Wavelet Transform using Medical Image

Ramesh Muthiya[1], Gomathy Balasubramanian[2], and Sundararajan Paramasivam[3]
[1]Department of Electronics and Communication Engineering, St.Martin's Engineering College, India
[2]Department of Computer and Science Engineering, Bannari Amman Institute of Technology, India
[3]Department of Electronics and Communication Engineering, Sri Shakthi Instituteof Engineering and Technology, India

Abstract: *Owing to the development of latest technologies in the areas of communication and computer networks, present day businesses are moving to the digital world for effectiveness, convenience and security. There are a number of applications in healthcare industry like tele-consulting, tele-surgery and tele-diagnosis. Today's healthcare involves some security risks as these provide new ways to store, access and distribute medical data. Watermarking can be seen as an additional tool for security measures. Pseudorandom noise sequence image watermarking algorithm which is blind (it does not require the presence of input image for detection) and robust is also analyzed. The watermarking scheme embeds the binary logo in the Discrete Wavelet Transform (DWT) domain as in the sub-band level. Consequently, the simulation results show that the proposed algorithm achieves higher security and robustness against various attacks like Set Partitioning in Hierarchical Trees (SPIHT) and JPEG compression, adding Gaussian noise and salt and pepper noise, Gaussian filtering and average filtering. The promising experimental results are Peak Signal-Noise Ratio (PSNR) and Normalized Correlation (NC) value is reported and also by using Compression Techniques (CT) scan and MRI medical images.*

Keywords: *Discrete wavelet transform, watermarking algorithm, pseudorandom noise sequence, peak signal-noise ratio, normalized correlation and medical images.*

Received September 22, 2014; accepted December 23, 2014

1. Introduction

A good deal of research has been done to increase the robustness and the data hiding capacity of watermarking techniques based on perceptual properties of the Human Visual System (HVS) (Kay and Izquierdo [3], Wolfgang *et al.* [9]). The development and improvement of accurate human vision models help in the design and growth of perceptual masks that can be used to better hide the watermark information thereby increasing its security.

There is a trade-off between robustness and imperceptibility (Swanson *et al.* [8], Cox and Miller [2]).

Most steganographic techniques that are designed to be robust must insert the watermark information into the cover image in a way that is perceptually significant. Other techniques that are relatively better at hiding information, like the Least Significant Bit (LSB) method, are highly vulnerable to having the embedded data distorted or quantized by lossy image compressions like Joint Photographic Experts Group (JPEG). For obvious reasons, it is imperative to consider an invisible watermarking method that is capable of hiding the watermark information in the cover image in an unnoticeable way. This imperceptibility is obtained by considering the various properties of the HVS that make the scheme more robust to many types of attacks. Existing algorithms for watermarking still images usually work either in spatial domain or in transformed domain Abu Sa'dah *et al.* [1].

The watermarking scheme is deals with the extraction of the watermark information in the absence of the original image, i.e., blind watermarking. Hence correlation-based watermark detection is used. A decimal sequence is added, to the cover object, instead of a PN sequence, based on the actual watermark. The results and formulae are based on a (512x512) size cover image and a block refers to a Discrete Cosine Transform (DCT) block of size (8x8), which is used for better robustness against JPEG compression.

2. PN Sequence Watermarking Algorithm

An improved Discrete Wavelet Transform (DWT) based Pseudorandom Noise (PN) sequence for watermarking (IDPW-Improved Discrete wavelet transform based PN sequence for Watermarking) algorithm using medical input image is proposed. The diamond shape of LH and HL sub-band decomposition is performed on the medical image using Haar wavelet transform. A Haar wavelet transform is conceptually simple and fast. It is exactly reversible without any edge effects. The watermark used in the algorithm is in signature image form. This watermark signature is embedded in the same transform coefficients of the input image using uncorrelated codes.

For each message bit, two different Pseudo Noise (PN) metrics namely size, identical to the size of the wavelet coefficient matrices, are generated. Since the security level of the watermarking algorithm depends on the strength of its secret key, a gray scale image is used as a strong key for generating pseudorandom sequences Ramesh *et al.* [7]. Based on the value of the bit for the message vector, the respective two PN sequence values are then added to the corresponding first level LH and HL coefficients values respectively according to the data embedding rule as follows:

$$W=V+kX \text{ } if \text{ } b=0 \quad (1)$$

Where V is wavelet coefficient of the host image, W is the wavelet coefficient after watermark embedding, k is the gain factor, X is the PN value and b is the bit of watermark that needs to be embedded. The generation of a pair of PN value for embedding each bit enhances the security of the watermarking algorithm. The following steps are applied in image embedding process for the proposed algorithm shown in Figure 1.

2.1. Embedding Process

The image I (M, N) of size MxN is used as the input.

- *Step 1.* The message to be hidden is read and is converted into binary sequences D_d (D_d=1 to n).
- *Step 2.* The input image is transformed using Haar Wavelet transform and first level sub-band coefficientscc A, ccH, ccV, ccD are obtained.
- *Step 3.* The n-different PN-sequence pairs (PN_h and PN_v) each of size (M/4)*(N/4) are generated using a secret key to reset the random number.
- *Step 4.* For D_d =1 to n, PN sequences to ccH and ccV components when message = 0 is added.
  - ccH = ccH + k*PN_h;
  - ccV = ccV + k*PN_v;

Where k is the gain factor used to specify the strength of the embedded data. Then an inverse Haar Wavelet transform is applied to get the final watermarked image I_w (M.N).

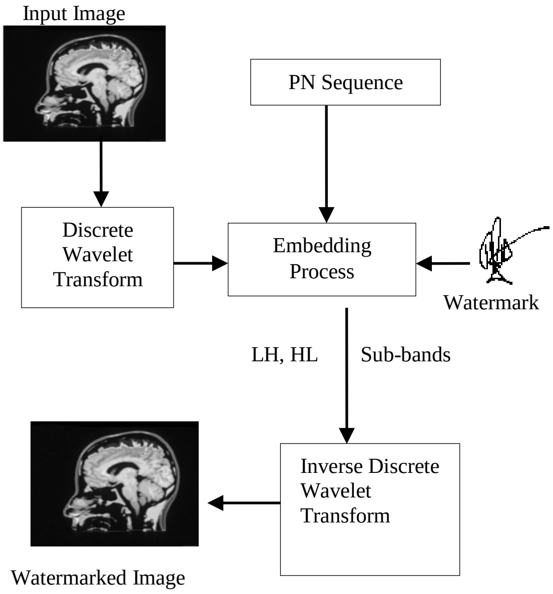

Figure 1. Embedding process.

2.2. Extraction Process

To detect the watermark the same pseudorandom values are generated and used during the insertion of watermark by using the same state key and its average correlation is determined with the two detail sub-bands DWT coefficients. Then an average of n correlation coefficients corresponding to each PN values is obtained for both LH and HL sub-bands. The mean of the average correlation values are taken as threshold T for message extraction. During detection, if the average correlation exceeds T for a particular sequence a “0” is recovered; otherwise a “1”. The recovery process then iterates through the entire PN sequence until all the bits of the watermark have been recovered. For extracting the watermark, the following steps are applied to the watermarked image shown in Figure 2.

- *Step 1.* The watermarked image I_w (M.N) is read.
- *Step 2.* The watermarked image using Haar wavelet transform is transformed and ccA1, ccH1, ccV1, ccD1 coefficients are obtained.
- *Step 3.* One’s sequences (msg) equal to message vector (from 1 to n) are generated.
- *Step 4.* The n-different PN-sequence pairs (PN_h1and PN_v1) each of size (M/4)*(N/4) is generated using the same secret key which is used in embedding to reset the random number generator.
- *Step 5.* For i=1 to n the correlations are calculated and these values are stored in corr_H (i) and corr_V (i).corr H(i)=correlation between PN_h1(i) and ccH1(i)corr V(i)=correlation between PN_v1(i) andccV1(i).

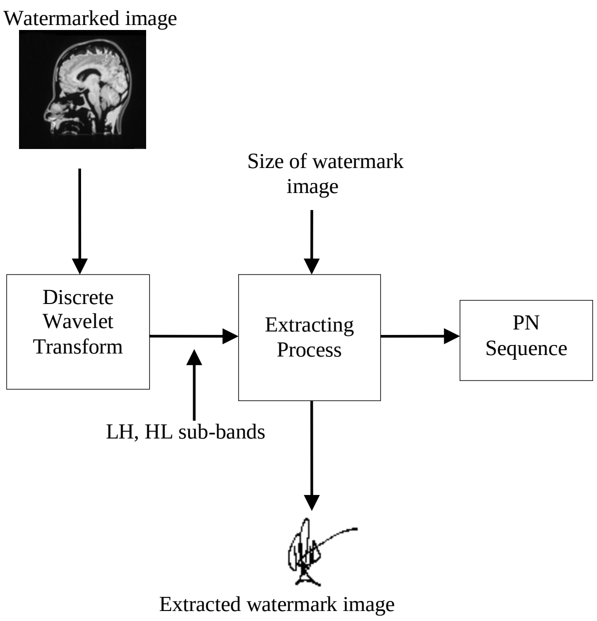

Figure 2. Extracting process.

- *Step 6.* Average correlation avg_ corr(i)=(corr_ H (i) +corr_V (i))/2 is calculated.
- *Step 7.* Thecorr(n) is calculated where corr(n)=mean of all the values stored in avg _ corr(i)
- *Step 8.* The hidden bit 0 is extracted, using the relationship given below For j=1 to nfavg_ corr(j) >corr(n), msg(j)=0
- *Step 9.* These extracted messages are rearranged.

3. Results and Discussion

Experiments are conducted using the different input CT scan image and watermark image. The size of the host image is 256×256 pixels. The size of the watermark image is 32×32 pixels. A Haar Wavelet filter is used for wavelet decomposition. The host image is decomposed into four sub-bands LL, LH, HL and HH. The watermark image is embedded in the LH and HL sub-bands.

Figure 1 shows the input CT scan image (Head) and watermarked images obtained by applying a watermarking algorithm in the first level LH and HL sub-band DWT coefficients at different compression ratio. The extracted watermarks along with the original watermarks are shown in Figure 2. The visual appearance of the watermarked image is good and shows no significant artefacts or distortions because of the process of watermarking. The Peak Signal to Noise Ratio (PSNR) between the input and the watermarked image is expressed in dB and indicates the energy of inserted watermark. The PSNR depends on the Mean Squared Error (MSE) which is calculated according to Equation (3) where ‘I_o’ and ‘I_w’ are the input and watermarked images, and M and N are image dimensions:

$$MSE=\frac{1}{N}\sum_{i=0}^{i=N-1}\sum_{j=0}^{j=M-1}[I_o(i,j)-I_w(i,j)]^2 \quad (2)$$

$$PSNR=10\log_{10}\frac{(255\times 255)}{MSE} \quad (3)$$

The normalized correlation is used as a metric to compare the robustness. After extracting the watermark, the Normalized Correlation Coefficient (NCC) is computed using the input watermark and extracted watermark to judge the existence of watermark.

$$NCC=\frac{\sum_{i=0}^{h-1}\sum_{j=0}^{w-1}[W_o(i,j)\bullet W_e(i,j)]}{h\bullet w} \quad (4)$$

Where, h and w are the height and width of the watermark respectively. $W_o(i,j)$ and $W_e(i,j)$ are the values located at coordinate (x,y) of the input watermark and the extracted watermark. Here $W_o(i,j)$ is set to 1 if it is a watermark bit 1; otherwise, it is set to -1.$W_e(i,j)$ is set in the same way. So the value of $W_o(i,j)\cdot W_e(i,j)$ is either -1 or 1

The graph in Figure 3 represents the comparison of PSNR value versus various compression ratios (0-100) in Set Partitioning in Hierarchical Trees (SPIHT) compression attacks for different watermarking algorithms. The IDPW algorithm begins with 24 dB and ends with 45 dB. The PSNR value is maximum, say 10dB for Liu *et al.* [6] algorithm. Till the compression ratio is 60, Ye and Tan [10] algorithm has the constant PSNR value 30 dB and after that again it is constant with the PSNR value 34 dB. Kung *et al.* [4] algorithm has different PSNR value between 20 dB to 33 dB. Lin *et al.* [5] has PSNR value of 34 dB only.

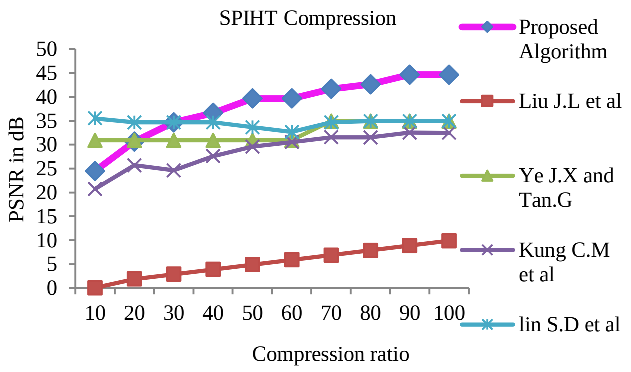

Figure 3. PSNR values obtained by the different algorithms for SPIHT compression.

For several watermarking techniques, Figure1 denotes the different compression ratios (0-100) with respect to the PSNR values. From Figure 3, it can be observed that for a compression ratio of 100, the IDPW algorithm shows 78%, 22%, 27%, and 22% improvement in PSNR value when compared with Liu *et al.* [6], Ye and Tan [10], Kung *et al.* [4] and Lin *et al.* [5] algorithms respectively.

The graph in Figure 4 conveys the comparison of PSNR value versus various compression ratios (0-100) in JPEG compression attacks for different watermarking algorithms. The IDPW algorithm begins with 29dB and ends with 45dB. The PSNR value is 15dB i.e., maximum for the Liu *et al*. [6] algorithm.

The PSNR value is initially 23dB and finally reaches 37dB for Ye and Tan [10] algorithm. Kung *et al*. [4] algorithm is initially 31dB high and finally reduces to 27db. Lin *et al*. [5] algorithm has above 30dB PSNR value.

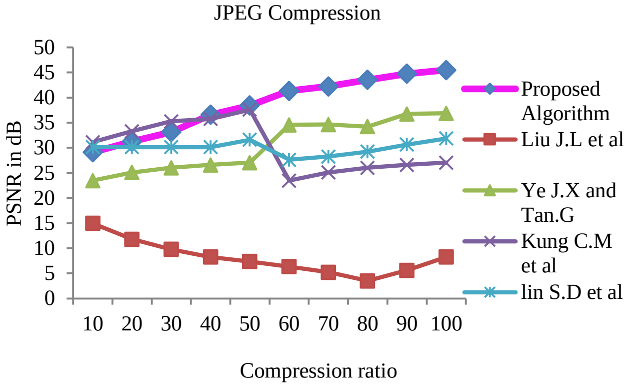

Figure 4. PSNR values obtained by the different algorithms for JPEG compression.

For several watermarking techniques, Figure 4 shows the different compression ratios (0-100) with respect to the PSNR values. From Figure 4, it can be observed that for a compression ratio of 100, the IDPW algorithm shows 81%, 19%, 40% and 30% improvement in PSNR value when compared with Liu *et al.* [6], Ye and Tan [10], Kung *et al.* [4] and Lin *et al.* [5] algorithms respectively.

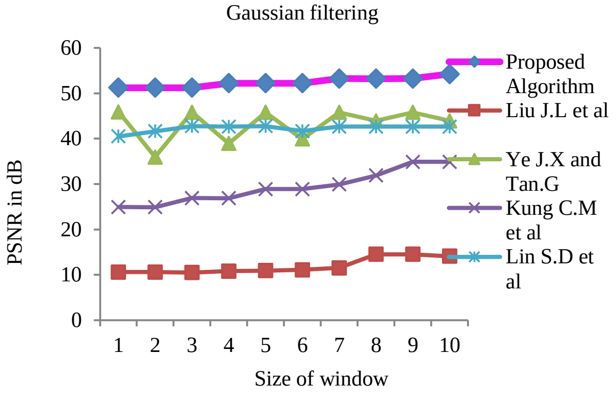

Figure 5. PSNR values obtained by the different algorithms for gaussian filtering.

The graph in Figure 5 deals with the comparison of PSNR value versus different sizes of window (1-10) in Gaussian filtering attacks for various watermarking algorithms. The IDPW algorithm has above 51dB PSNR value. The PSNR value is between 10-14dB for Liu *et al.* [6] algorithm. The PSNR value varies till 45dB for Ye and Tan [10] algorithm. Kung *et al.* [4] algorithm has PSNR value varying from 25dB to 35dB. Lin *et al.* [5] algorithm has constant PSNR value of 42dB.

For several watermarking techniques, Figure 5 shows the different sizes of window (1-10) with respect to the PSNR values. Figure 2 ensures that for a size of window (7x7), the IDPW algorithm shows 74%, 14%, 35% and 22% increase in PSNR value when compared with Liu *et al.* [6], Ye and Tan [10], Kung *et al.* [4] and Lin *et al.* [5] algorithms respectively.

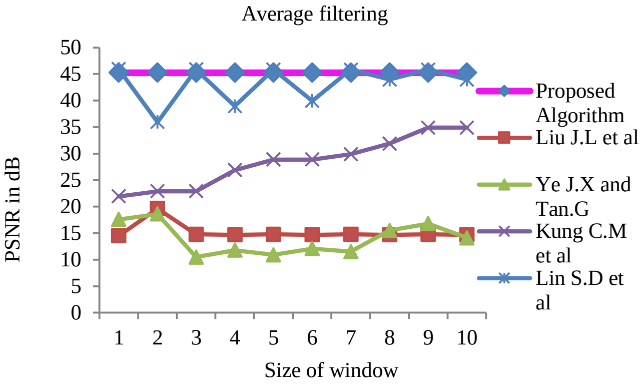

Figure 6. PSNR obtained value by the different algorithms for averaging filtering.

The graph in Figure 6 draws attention to the comparison of PSNR value versus different size of windows (1-10) in averaging filtering attacks for various watermarking algorithms. The IDPW algorithm has constant PSNR value of 45dB. The PSNR value is 14dB constant for the Liu *et al.* [6] algorithm. Ye and Tan [10] algorithm has PSNR value initially of 17dB and finally reduces to 14dB. Kung *et al.* [4] algorithm has PSNR value varying from 22dB to 35dB. The PSNR value 45dB is constant for Lin *et al.* [5] algorithm.

For several watermarking techniques, Figure 6 shows the different size of windows (1-10) with respect to the PSNR values. Figure 6 proves that for a size of window (3x3), the IDPW algorithm shows 67%, 77%, 49% and 22% increase in PSNR value when compared with Liu *et al.* [6], Ye and Tan [10] and Kung *et al.* [4] algorithms respectively and then it is reduced 1.3% PSNR value when compared with Lin *et al.* [5] algorithm.

The graph in Figure 7 contains the comparison of PSNR value versus various noise densities (0-1) in Gaussian noise attacks for different watermarking algorithms. The IDPW algorithms have more 60dB PSNR value. The PSNR value is initially 24dB and finally improves to 32dB for Liu *et al.* [6] algorithm.

Ye and Tan [10] algorithm has a marginal reduction in PSNR value from 45dB to 41 dB. Kung *et al.* [4] algorithm has constant PSNR value 38dB. Lin *et al.* [5] algorithm has PSNR value varying from 29dB to 49dB.

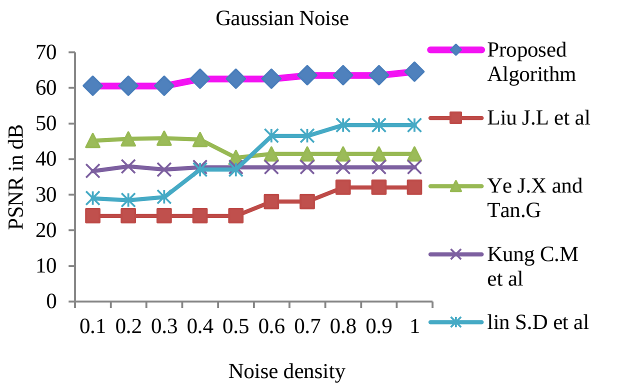

Figure 7. PSNR value obtained by the different algorithms for gaussian noise.

For several watermarking techniques, Figure 7 shows the various noise densities (0-1) with respect to the PSNR values. Figure 7 informs that for a noise density 0.8, the IDPW algorithm shows 49%, 35%, 40% and 22% raise in PSNR value when compared with Liu *et al*. [6], Ye and Tan [10], Kung *et al*. [4] and Lin *et al*.[5] algorithms respectively.

The graph in Figure 8 shows the comparison of PSNR value versus various noise densities (0-1) in salt and pepper noise attacks for different watermarking algorithms. The IDPW algorithm PSNR value begins with 36dB and ends with 60dB. The PSNR value is initially 46dB and finally reduces to 40dB for Liu *et al.* [6] algorithm. Ye and Tan [10] algorithm has a constant PSNR value of 45dB. The PSNR value varies from 17dB to 30dB for Kung *et al.* [4] algorithm. Lin *et al.* [5] algorithm has initially 27dB PSNR value and finally improves to 48dB.

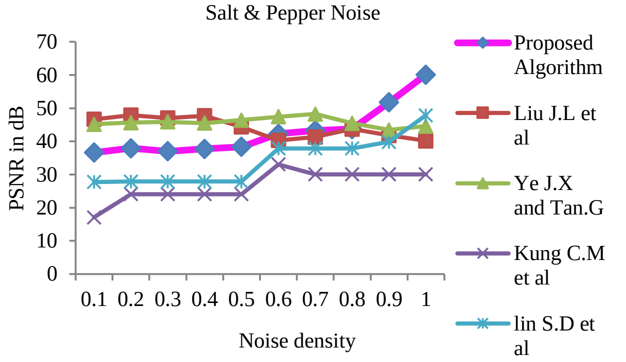

Figure 8. PSNR value obtained by the different algorithms for salt and pepper noise.

For several watermarking techniques, Figure 8 shows the various noise densities (0-1) with respect to the PSNR values. Figure 8 confirms that for a noise density 0.8, the IDPW algorithm shows 33%, 26%, 50% and 20% raise in PSNR value when compared with Liu *et al*. [6], Ye and Tan [10], Kung *et al*.[4] and Lin *et al*.[5] algorithms respectively.

The graph in Figure 9 expresses the comparison of NC value versus various compression ratios (0-100) in SPIHT compression attacks for different watermarking algorithms. The constant NC values such as 0.96, 0.76 and 0.86 are obtained for the IDPW algorithm, Liu *et al.* [6] and Lin *et al.* [5] algorithms respectively.

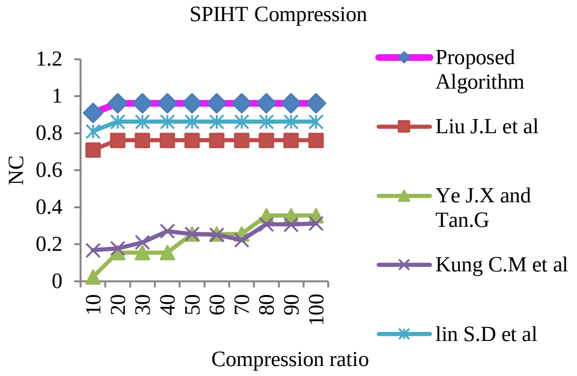

Figure 9. Normalized correlation value obtained by the different algorithms for SPIHT compression.

Ye and Tan [10] algorithm has initially very low NC value 0.02 and slowly increases with 0.35 NC value. The NC value is varied from 0.16 to 0.31 for Kung *et al.* [4] algorithm.

For several watermarking techniques, Figure 9 shows the different compression ratios (0-100) with respect to the NC values. Figure 9 validates that for a compression ratio of 80, the IDPW algorithm shows 21%, 63%, 67% and 10% increase in NC value when compared with Liu *et al.* [6], Ye and Tan [10], Kung *et al.* [4] and Lin *et al.* [5] algorithms respectively.

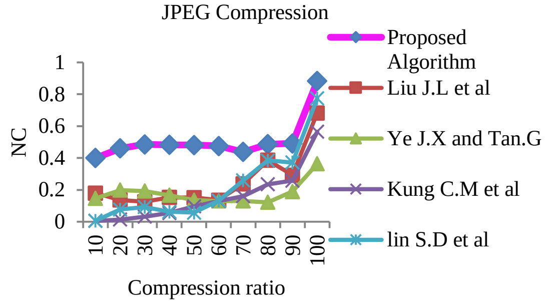

Figure 10. Normalized correlation value obtained by the different algorithms for JPEG compression.

The graph in Figure 10 highlights the comparison of NC value versus various compression ratios (0-100) in JPEG compression attacks for different watermarking algorithms. The IDPW algorithm is initially 0.4 and ends with the NC value 0.88. The NC value is varied from 0.18 to 0.68 for Liu *et al.* [6] algorithm. Ye and Tan [10] algorithm has constant 0.14 NC value till the compression ratio is 90 and then reaches to maximum at the end. The NC value is initially very low and finally high at the end for Kung *et al.* [4] and Lin *et al.* [5] algorithms.

For several watermarking techniques, Figure 10 shows the different compression ratios (0-100) with respect to the NC values. From Figure 10 it can be observed that for a compression ratio of 100, the IDPW algorithm shows 23%, 59%, 36% and 12% increase in NC value when compared with Liu *et al.* [6], Ye and

Tan [10], Kung *et al.* [4] and Lin *et al.* [5] algorithms respectively.

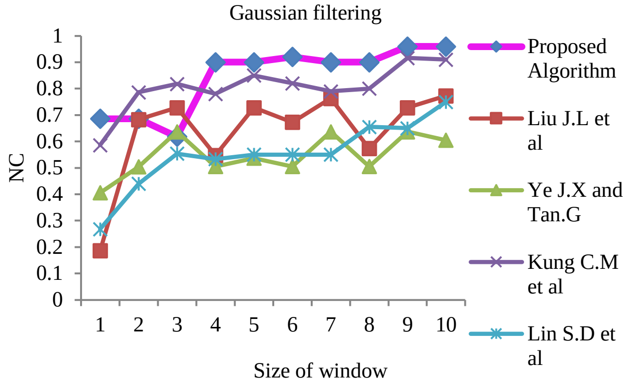

Figure 11. Normalized correlation value obtained by the different algorithms for gaussian filtering.

The graph in Figure 11 points out the comparison of NC value versus different sizes of windows (1-10) in Gaussian filtering attacks for various watermarking algorithms. The IDPW algorithm has always NC value above 0.9. In Liu *et al.* [6] algorithm, the NC value is initially very low 0.18 and finally increases by 0.77. Ye and Tan [10] algorithm shows varying NC value. The NC value is varied from 0.58 to 0.91 for Kung *et al.* [4] algorithm. Lin *et al.* [5] algorithm has NC value 0.55 up to the size of window (7x7) and the value is high at the end.

For several watermarking techniques, Figure 11 shows the different sizes of windows (1-10) with respect to the NC values. Figure 11 makes it clear that for a size of window (5x5), the IDPW algorithm shows 20%, 41%, 6% and 39% increase in NC value when compared with Liu *et al.* [6], Ye and Tan [10], Kung *et al.* [4] and Lin *et al.* [5] algorithms respectively.

The graph in Figure 12 shows the comparison of NC value versus different size of windows (1-10) in averaging filtering attacks for various watermarking algorithms. The IDPW algorithm has high NC value 0.78 initially and low 0.28 finally. For all the algorithms namely Lin *et al.* [5], Kung *et al.* [4], Ye and Tan [10] and Liu *et al.* [6], the NC value is high at the beginning and gradually becomes constant such as 0.13, 0.2, 0.11 and 0.25 respectively.

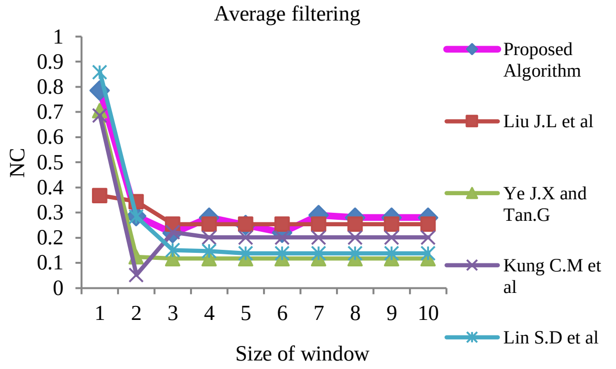

Figure 12. Normalized correlation value obtained by the different algorithms for average filtering.

For several watermarking techniques, Figure 12 shows the different size of windows (1-10) with respect to the NC values. Figure 12 endorses that for a size of window (7x7), the proposed IDPW algorithm shows 11%, 61%, 28% and 53% improvement in NC value when compared with Liu *et al.* [6], Ye and Tan [10], Kung *et al.* [4] and Lin *et al.* [5] algorithms respectively.

The graph in Figure.13 presents the comparison of NC value versus various noise densities (0-1) in Gaussian noise attacks for different watermarking algorithms. In the IDPW algorithm, the NC value begins with 0.28 and ends with 0.92. Lin *et al.* [5] algorithm has a constant NC value 0.92 up to the noise density 0.6 and then it is reduced to 0.72. Kung *et al.* [4] algorithm has maximum NC value up to 0.7 noise densities and after that it is reduced to 0.57. The NC value is initially 0.48 and finally increased to 0.89 for Ye and Tan [10] algorithm. The NC value is increased in three stages for Liu *et al.* [6] algorithm.

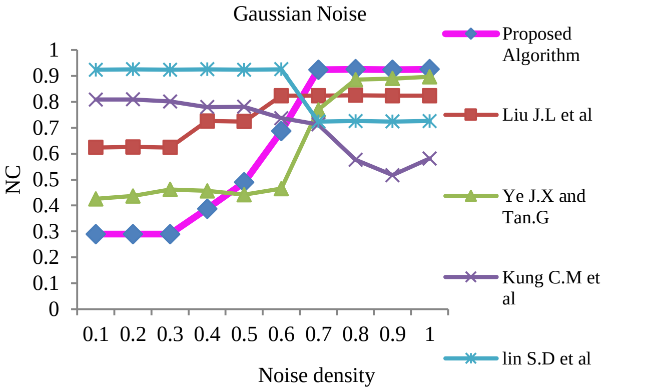

Figure 13. Normalized correlation value obtained by the different algorithms for gaussian noise.

For several watermarking techniques, Figure 13 shows the various noise densities (0-1) with respect to the NC values. Figure 13 makes it clear that for a noise density 0.7, the proposed IDPW algorithm shows 11%, 17%, 23% and 22% raise in NC value when compared with Liu J L *et al.* [6], Ye and Tan [10], Kung *et al.* [4] and Lin *et al.* [5] algorithms respectively.

The graph in Figure 14 provides the comparison of NC value versus various noise densities (0-1) in salt and pepper noise attacks for different watermarking algorithms. The IDPW algorithm has almost the same NC value of 0.98. Lin *et al.* [5] algorithm has constant NC value 0.92 till 0.6 noise density and after that again it becomes constant for the value 0.98. Kung *et al.* [4] algorithm has three stage increases in NC value. The NC value 0.8 is initially high and finally reduced to 0.51 for Ye and Tan [10] algorithm. In Liu *et al.* [6] algorithm the NC value is varied from 0.48 to 0.89.

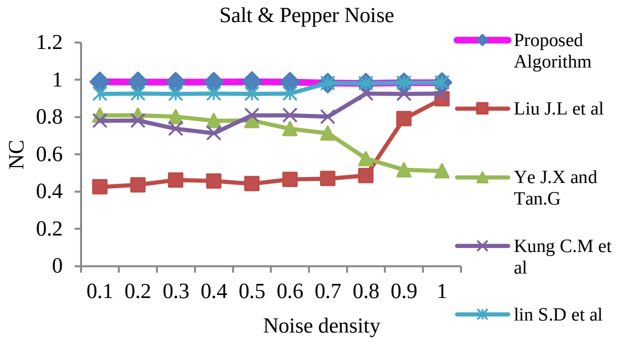

Figure14. Normalized correlation value obtained by the different algorithms for salt and pepper noise.

For several watermarking techniques, Figure 14 shows the various noise densities (0-1) with respect to the NC values. Figure 14 reveals that for a noise density 0.1, the IDPW algorithm shows 57%, 18%, 20% and 6% increase in NC value when compared with Liu *et al.* [6], Ye and Tan [10], Kung *et al.* [4] and Lin *et al.* [5] algorithms respectively.

4. Conclusions

In this paper, an improved discrete wavelet transform based PN sequence for watermarking (IDPW) algorithm has been discussed. The proposed IDPW algorithm has telemedicine applications by watermarking radiological images with sensitive medical information in binary image format. Medical information such as telemedicine origin centre of doctor's signature (watermark) is embedded into input CT scan image as watermarks. These watermarks are in binary image formats which add robustness by allowing recovery of the watermarks even at low correlation between original and extracted watermarks.

In the IDPW algorithm, for both the Gaussian and average filtering attacks of all sizes of window 1to10, the Peak Signal-Noise Ratio value is constant and with 22% improvement when compared with both Kung *et al.* [4] and Lin *et al.* [5] algorithms. Similarly, when adding Gaussian noise attacks for all noise densities 0 to 1, the PSNR value stays constant and shows 22% improvement when compared with Lin *et al.* [5] algorithm.

In regard to the SPIHT compression attacks for all compression ratios 0 to 100 the normalized correlation value is constant at 0.96 and 10% increase when compared with Lin *et al.* [5] algorithm. Similarly, in salt and pepper noise attacks for all noise densities 0 to1, the normalized correlation value is constant at 0.98 and raises 6% higher when compared with Lin *et al.* [5] algorithm. When embedding as well as extraction phases are performed in wavelet domain, the IDPW approach is highly secure and it is robust to all sorts of watermarking attacks. The robustness of the IDPW algorithm is good against all types of attacks. Hence the IDPW algorithm is suitable for copyright protection, potential applications in military, medical and law enforcement related image processing.

References

[1] Abu Sa'dah Y., Al-Najdawi N., and Tedmori S., "Exploiting Hybrid Methods for Enhancing Digital X-Ray Images," *The International Arab Journal of Information Technology*, vol. 10, no. 1, pp. 28-35, 2013.

[2] Cox I. and Miller M., "A Review of Watermarking and the Importance of Perceptual Modelling," in Proceedings of the Conference on Human Vision and Electronic Imaging, San Jose, pp. 92-99, 1997.

[3] Kay S. and Izquierdo E., "Robust Content Based Image Watermarking," *in Proceedings of Workshop on Image Analysis for Multimedia Interactive Services*, Finland, 2001.

[4] Kung C., Chao S., Tu Y., Yan Y., and Kung C., "A Robust Watermarking and Image Authentication Scheme used for Digital Content Application," *Journal of Multimedia*, vol. 4, no. 3, pp. 112-119, 2009.

[5] Lin S., Shih S., and Guo J., "Improving the Robustness of DCT-based Image Watermarking Against JPEG Compression," *Computer Standards and Interfaces*, vol. 32, no. 1-2, pp. 54-60, 2010.

[6] Liu J., Lou D., Chang M., and Tso H., "A Robust Watermarking Scheme using Self-Reference Image," *Computer Standards and Interfaces*, vol. 28, no. 3, pp. 356-367, 2006.

[7] Ramesh S., Shanmugam A., and Gomathy B., "Implementation of Watermarking for a Blind Image using Wavelet Tree Quantization," *International Journal of Computer and Network Security*, vol. 2, no. 3, pp. 50-55, 2010.

[8] Swanson M., Kobayashi M., and Tewfik A., "Multimedia Data-embedding and Watermarking Techniques," *in Proceedings of IEEE*, vol. 86, no. 6, pp. 1064-1087, 1998.

[9] Wolfgang R., Podilchuk C., and Delp E., "Perceptual Watermarks for Digital Images and Video," *in Proceedings of the IEEE*, vol. 87, no. 7, pp. 1108-1126, 1999.

[10] Ye J. and Tan G., "An Improved Digital Watermarking Algorithm for Meaningful Image," *in Proceedings of International Conference on Computer Science and Software Engineering*, Hubei, pp. 822-825, 2008.

Ramesh Muthiya received the B.E degree in Electronics and Communication Engineering from National Institute of Technology (Formerly Regional Engineering College), Trichy, Bharathidhasan University, India in the year 2001 and the M.E, degree in Applied Electronics from R.V.S. College of Engineering and Technology, Dindugal, Anna University, India in the year 2004. He has completed his Ph.D. degree in Image processing from Anna University, Chennai, India in the year 2012. Currently, he is working as Professor in the Department of Electronics and Communication Engineering, St.Martin's Engineering College, Hyderabad and also 2 years worked as Professor in the Department of Electronics and Communication Engineering, E.G.S Pillay Engineering College, Nagapattinam and also 10 years worked as Associate Professor in the Department of Electronics and Communication Engineering, Bannari Amman Institute of Technology, Sathyamangalam, India. His current research focuses on Image Processing, Low Power VLSI Design and Soft Computing. He has published 74 articles in National and International Journals and more than 28 papers in International and National Conferences. He is life member of ISTE and IETE.

Gomathy Balasubramanian received the B.E degree in Computer Science and Engineering from Sethu Institute of Technology, Madurai, Madurai Kamaraj University, India in the year 1999 and the M.E, degree in Computer Science and Engineering from Bannari Amman Institute of Technology, Sathyamangalam, Anna University, India in the year 2009. She has completed her Ph.D. degree in Data Mining from Anna University, Chennai, India in the year 2014. Currently, she is working as Associate Professor in the Department of Computer and Science Engineering, Bannari Amman Institute of Technology, Sathyamangalam, India. She has published 34 articles in National and International Journals and more than 22 papers in International and National Conferences. She is member of ISTE.

Sundararajan Paramasivam received the B.E. Degree in Electronics and Communication from Kongu Engineering College, Perundurai, India in 1993 and the M.E. Degree in Applied Electronics from the Government College of Technology, Coimbatore, India in 1999. He has completed his Ph.D. in Anna University, Chennai in the area of Security in Mobile ad-hoc Networks. Currently, he is working as Professor in the Department of Electronics and Communication Engineering, Sri Shakthi Institute of Engineering &Technology, Coimbatore and also 18 years worked he is working as Professor in the Department of Electronics and Communication Engineering, Bannari Amman Institute of Technology, Sathyamangalam, India. He has published 63 articles in National and International Journals and more than 36 papers in International and National Conferences. His current research focuses on mobile ad hoc networks and wireless security. He is member of IE and ISTE.