An Approach towards Designing a Robust DCT based Image Watermarking Framework against JPEG Compression

Soumik Das
MCA Department, Techno India
Salt Lake City, Kolkata, India

Monalisa Banerjee
MCA Department, Techno India
Salt Lake City, Kolkata, India

Atal Chaudhuri
Computer Sc. & Engg. Dept.
Jadavpur University
Kolkata, India

ABSTRACT
Ownership authentication and copyright protection of image has achieved a sharp attention due to easy and global availability of internet services. Illegal copying and misappropriation of digital image leads cyber crime in an epidemic form. Invisible watermarking technique is one of the leading solutions towards violation of ownership authentication problem without deteriorating the quality of the image. In spatial domain watermarking, a counterfeiter may not retrieve the watermark from watermarked image but the watermark can be destroyed if JPEG compression is performed on the watermarked image even at a very low level of compression and this is a very popular and easy way indeed for violating authentication of ownership. In this article a DCT based image watermarking framework is proposed to enhance the robustness of the watermark in the watermarked image against high level lossy JPEG compression. Several proposed watermark frameworks in last few years have considered binary watermarks and watermark pixels are directly embedded at the DCT coefficients of host images. Whereas in our proposed framework we have used color host images and grayscale watermarks and DCT is performed on both the host image and watermark image. Watermark frequencies are embedded in the DCT coefficients of several blocks of the host image. A secret key is used that determines the embedding blocks of the host image. Experimental result sets demonstrate that our proposed framework ensures the quality of watermarked image, perceptual invisibility of watermark in human visual system, robustness of watermark against high JPEG compression.

General Terms
Image security, Digital watermarking, Copyright protection

Keywords
Ownership, authentication, invisible watermarking, DCT, JPEG compression

1. INTRODUCTION
Digital image watermarking is a method for embedding information into a digital image. The image watermarking technique provides a persistent connection between the authenticator and the image it authenticates [1]. The digital image watermarking (hereinafter referred to as watermarking for rest of the paper) can be categorized into different types— visible and invisible as well as spatial domain and frequency domain [2]. In current communication we've focused in invisible frequency domain watermarking. Our already proposed different frameworks have been proven efficient to embed color watermark into color host images in spatial domain [3-6]. If the watermarking is done with these frameworks a counterfeiter will not be able to retrieve the watermark from watermarked image without having a proper secret key but even if a low level JPEG compression is performed on the watermarked image then ownership will be seriously destroyed. To overcome these limitations we've designed a DCT based watermarking framework that can embed a gray scale watermark into a color host image. The color host image will be transformed from RGB color space to YUV color space because RGB color space is highly correlated and not suitable for frequency domain watermarking applications [7]. After that Y part of the host image will be divided into 8X8 non-overlapping blocks and DCT will be performed on each block. To improve the robustness of the watermark against JPEG compression, the DCT transformed blocks of Y are quantized. At the same time DCT will also be done on the grayscale watermark. A secret key of dynamic length (depending on the size of the watermark) will determine the embedding blocks through proposed algorithm. And then watermark intensities will be inserted at the quantized DCT blocks. Performing de-quantization and inverse DCT will give the watermarked image. At the extraction end the same key must be provided to determine the blocks where the watermark is embedded. If a counterfeiter tries to extract the watermark with an improper key, the original watermark will not be constructed. The key aesthetics of the proposed watermarking system are—

- Quality of watermarked image is preserved.
- Watermark is perceptually invisible in HVS.
- Highly robust against JPEG compression.
- Use of a dynamic length secret key that determines embedding blocks.

In current communication we've discussed the framework and algorithmic approach for watermark embedding and extraction in Section 2, experimental result set is reported at Section 3, conclusion is being made at Section 4 and references are given at Section 5.

2. FRAMEWORK & ALGORITHMIC APPROACH
We've portrayed the algorithm for embedding and extraction of watermark in following three ways for better understanding— function based algorithmic approach, detailed stepwise approach and schematic diagram. The next subsection will elaborate the embedding algorithm.

2.1 Watermark Embedding Algorithm

The function based algorithm is depicted below—

[Watermark Embedding Algorithm]

- Input: Color Host Image (H), Grayscale Watermark (A), Secret Key (K)
- Output: Color Watermarked Image (W)

Step1. H_{YUV} = *Transform YUV* (H_{RGB})

Step2. H_{Y} = *Separate Y* (H_{YUV})

Step3. H_{YDCT} = *Perform FDCT* (H_{Y})

Step4. $H_{Y\text{-}QUANTIZED}$ = *Quantize* (H_{YDCT})

Step5. A_{DCT} = *Perform FDCT (A)*

Step6. $BLOCKS_{SELECTED}$ = *Determine Blocks FDCT* (K, H_{YDCT})

Step7. $W_{EMBEDDED}$ = *Embed* (A_{DCT} , $BLOCKS_{SELECTED}$)

Step8. $W_{DEQUANTIZED}$ = *Dequantize* ($W_{EMBEDDED}$)

Step9. W_{YUV} = *Perform IDCT* ($W_{DEQUANTIZED}$)

Step10. W_{RGB} = *Transform RGB* (W_{YUV})

The detailed stepwise approach for watermark embedding algorithm is as follows—

2.1.1 Transformation from RGB to YUV:

The host image will be transformed from RGB color space to YUV color space because RGB color space is highly correlated and not suitable for performing the DCT [8]. Y part is called the luminance component where as the U and V part is called the chrominance. Although the luminance is much sensitive to HVS (Human Visual System) than the chrominance but also the luminance (Y) channel of host image is used to embed watermark because JPEG compression measurably changes the chrominance part of an image. So if watermark insertion is done at chrominance part then after JPEG compression the watermark will not be preserved. The transformation from RGB color space to YUV color space is done with following Equation1 [7].

$$\begin{pmatrix} Y \\ U \\ V \end{pmatrix} = \begin{pmatrix} 0.299 & 0.587 & 0.114 \\ -0.148 & -0.289 & 0.437 \\ 0.615 & -0.515 & -0.100 \end{pmatrix} \times \begin{pmatrix} R \\ G \\ B \end{pmatrix} \qquad (1)$$

2.1.2 Forward DCT of Y part of host image:

Discrete Cosine Transform (DCT) is one of the poplar ways to convert an image from spatial domain to frequency domain. The DCT works by separating image into parts of differing frequencies. The forward DCT of an image will be achieved from Equation2 [8].

$$DCT\,(i,j) = C(i)C(j)\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} pixel\,(x,y)\cos\left[\frac{(2x+1)i\pi}{2N}\right]\cos\left[\frac{(2y+1)j\pi}{2N}\right] \qquad (2)$$

where

$$C(i), C(j) = \begin{cases} \sqrt{\frac{1}{N}} & for\ i,j = 0 \\ \sqrt{\frac{2}{N}} & for\ i,j = 1,2,3 \ldots N-1 \end{cases}$$

p(x, y) is the x,yth element of the image represented by matrix p. N is the size of the block on that the DCT is done. Equation2 determines one entry (i,jth) of the transformed image from the pixel values of the original image matrix. In our proposed framework the luminance part Y of the host image is divided into 8X8 (N=8) non-overlapping blocks and forward DCT is performed on individual such blocks.

2.1.3 Quantization:

After performing DCT on all 8X8 blocks of host image we have done quantization based on standard quantization table Q_{50} for JPEG compression at quality level 50[8]. Quantizing DCT blocks improve the robustness of watermark against JPEG compression and help us to locate the higher frequencies which are less sensitive to HVS and where the watermark frequencies will be embedded. Quantization will be achieved by dividing each element in the transformed image matrix DCT (i, j) by the corresponding element in the quantization matrix Q (i, j) with rounding off to next integer value. Quantization will be done with Equation3 [8].

$$Quantized\_Block\ (i,j) = round - off\left(\frac{DCT\,(i,j)}{Q_{50}}\right) \qquad (3)$$

After applying DCT on a 8X8 block of an image Vegetable.jpg we got DCT (i, j) as follows.

$$DCT\ (i,j) = \begin{bmatrix} 150.1 & 30.7 & 20.0 & 71.0 & 24.2 & 26.0 & -10.5 & -20.3 \\ 22.0 & 79.3 & 20.5 & 28.7 & 17.5 & -10.6 & 13.6 & -10.0 \\ -76.8 & 11.6 & -56.0 & 32.3 & -21.2 & -6.7 & 1.6 & -9.7 \\ -28.3 & -66.5 & -7.6 & -17.2 & -23.5 & 12.6 & -2.3 & 10.2 \\ -30.1 & 19.3 & -7.3 & -15.0 & 19.7 & -8.1 & 31.0 & -11.0 \\ -0.7 & -13.7 & 17.5 & 0.6 & 45.3 & -9.4 & 11.2 & 1.3 \\ 14.5 & -5.6 & 20.0 & 7.7 & -45.3 & -13.1 & 3.8 & 25.1 \\ -10.1 & 32.0 & 19.0 & 21.5 & 33.0 & 9.8 & 0.0 & 7.2 \end{bmatrix}$$

Quantized_Block (i, j) will be achieved by Equation3 as follows.

$$Quantized\_Block\ (i,j) = \begin{bmatrix} 9 & 3 & 2 & 4 & 1 & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ 2 & 7 & 1 & 2 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ -5 & 0 & -4 & 1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ -2 & -4 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ -2 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}$$

2.1.4 DCT of watermark:

The grayscale watermark is divided into non-overlapping 8X8 blocks. And DCT is performed on each block using Equation2. The frequencies present in each 8X8 DCT blocks will be distributed and kept in two 8X8 blocks in such a way where half of the frequency values will be stored in individual blocks. Say there is a frequency value 90 in (0, 0) position of one 8X8 DCT block, and then it will be stored as 45 in (0, 0) position of two 8X8 blocks. We've adopted this scheme to ensure better invisibility of watermark in watermarked image.

2.1.5 Choosing watermark embedding blocks:

The length of the secret key is determined first depending on the size of the watermark. In Step4 we've divided our watermark in non-overlapping 8X8 blocks and DCT has been performed on them individually. So we know the number of non-overlapping 8X8 blocks is present in the grayscale watermark. If there is 2^n number of blocks is present then secret key will be of 2^{n+1} bit long. The length of the secret key is determined through Equation4 and 5.

$$Number\ of\ Blocks = \frac{N\times N}{8\times 8} = 2^n \quad (4)$$

Where size of the watermark is (NxN)

$$Length\ [secret\ key] = 2^{n+1}\ (bit) \quad (5)$$

For example let us consider a watermark of size 64X64 pixels. Length of the secret key is calculated below through Equation4 and 5.

$$Number\ of\ Blocks = \frac{64\times 64}{8\times 8} = 64 = 2^6$$

$$Length\ [secret\ key] = 2^{6+1} = 128\ bit$$

Now a 2^{n+1} bit long binary key will be chosen and embedding blocks will be determined through following algorithm.

- Algorithm for selecting embedding blocks

— Input {Total number of 8X8 blocks (H) in host image, Binary secret key (K)}
— Output {Embedding blocks of host image}

Step1. $K_{REVISED}$ = *Split Nibbles (K)*
Step2. $D_{SEQUENCE}$ = *Convert Decimal* ($K_{REVISED}$)
Step3. $ND_{SEQUENCE}$ = *Normalize* ($D_{SEQUENCE}$)
Step4. $DIGIT_{PLACE}$ = *Split Digit* ($ND_{SEQUENCE}$)
Step5. $BLOCKS_{SELECTED}$ = *floor* (H / $DIGIT_{PLACE}$) = {X, (X-1)} {Y, (Y-1)}…

For example, the 2^{n+1} bit key K=1001 1101…

[$K_{REVISED}$ = 1001 | 1101| …]
[$D_{SEQUENCE}$ = 9 | 13| …]
[$ND_{SEQUENCE}$ = 9, (1+3)4 …]
[$DIGIT_1$ = 9, $DIGIT_2$ = 4, …]
[$BLOCKS_{SELECTED}$ = *floor* (256/9), (256/4) = {28, 27},{64, 63} …]

If a collision occurs then next two available blocks will be selected through linear probing. The total block requirement for embedding will be twice the blocks present in the watermark. So if there is 2^n blocks present in the water mark, our requirement will be 2^{n+1}. A binary secret key of length 2^{n+1} is able to select 2^n blocks and same number of remaining blocks will be selected by executing the aforesaid algorithm with the secret key (K) in reverse order.

In previous step we observed how the algorithm determines the blocks where the embedding will be done. And we can see quantization process (Step3) gives us lots of zero elements in quantized blocks of host image. These are our desired high frequency positions where we intend to embed the watermark frequencies i.e., the DCT coefficients of watermark will replace the zero elements of the quantized blocks of host image. High frequencies will be selected sequentially from quantized blocks and DCT coefficients of individual block of watermark will be distributed sequentially into quantized blocks of host image determined by embedding block selection algorithm. This distribution technique will ensure the invisibility of watermark in the watermarked image.

2.1.6 De-quantization:

After embedding watermark frequencies into quantized host image block, we need to de-quantize the watermarked block.

2.1.7 Inverse DCT (IDCT) of watermarked blocks:

Inverse DCT needed to be performed on individual blocks after successful embedding of watermark frequencies in the host image blocks, IDCT will be done according to Equation6 [8].

$$pixel\ (x,y) = C(i)C(j)\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} DCT(i,j)\ cos\left[\frac{(2x+1)i\pi}{2N}\right] cos\left[\frac{(2y+1)j\pi}{2N}\right] \quad (6)$$

Where

$$C(i), C(j) = \begin{cases} \sqrt{\frac{1}{N}} & for\ i,j = 0 \\ \sqrt{\frac{2}{N}} & for\ i,j = 1,2,3 \ldots N-1 \end{cases}$$

2.1.8 Reverse YUV transform:

Finally the watermarked image will be transformed from YUV color space to RGB color space by using Equation7.

$$\begin{pmatrix} \mathbf{R} \\ \mathbf{G} \\ \mathbf{B} \end{pmatrix} = \begin{pmatrix} \mathbf{1} & \mathbf{0} & \mathbf{1.13983} \\ \mathbf{1} & \mathbf{-0.39465} & \mathbf{-0.58060} \\ \mathbf{1} & \mathbf{2.03211} & \mathbf{0} \end{pmatrix} \times \begin{pmatrix} \mathbf{Y} \\ \mathbf{U} \\ \mathbf{V} \end{pmatrix}$$
(7)

Fig.1 is depicted here to represent watermark embedding framework.

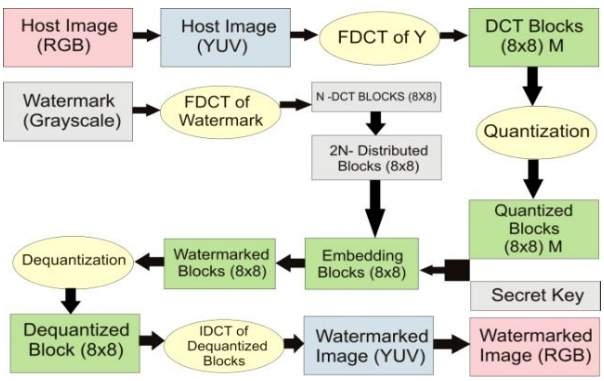


Fig. 1. Schematic Diagram of Embedding Framework

2.2 Watermark Extraction Algorithm

The function based algorithm is reported below—

[Watermark Extraction Algorithm]

- Input: Color Watermarked Image (W), Secret Key (K)
- Output: Grayscale Watermark (A)

Step1. W_{YUV} = *Transform YUV* (W_{RGB})
Step2. W_Y = *Separate Y* (W_{YUV})
Step3. W_{YDCT} = *Perform FDCT* (W_Y)
Step4. $BLOCKS_{SELECTED}$ = *Determine Blocks FDCT* (K, W_{YDCT})
Step5. $A_{EXTRACTED}$ = *Extract* ($BLOCKS_{SELECTED}$)
Step6. A = *Perform IDCT* ($A_{EXTRACTED}$)

The stepwise approach for watermark extraction is written hereinafter.

2.2.1 *Transformation from RGB to YUV:*

The color watermarked image needed to be transformed from RGB color space to YUV color space through Equation1 because we've done the watermarking at luminance (Y) part of the host image.

2.2.2 *Forward DCT of Y part of watermarked image:*

DCT will be performed on the Y part with the help of Equation2.

2.2.3 *Identification of embedding blocks:*

The same secret key (K) is to be applied on the watermarked image to identify the blocks where watermark frequencies have been embedded during embedding procedure.

2.2.4 *Formation of watermark:*

The watermark frequencies will be collaborated from identified blocks in order to form the grayscale watermark.

2.2.5 *Inverse DCT of formed watermark:*

In order to final formation of the grayscale watermark, it is divided into 8X8 non-overlapping blocks and Inverse DCT is performed on them individually by Equation6.
Fig.2 is depicted here to represent watermark extraction framework.

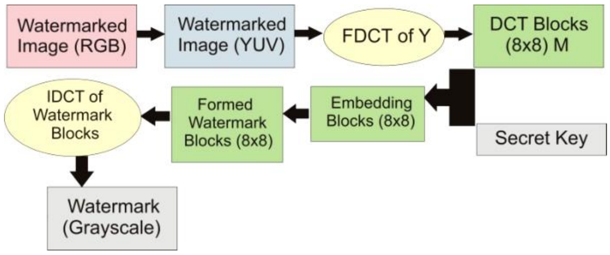


Fig. 2. Schematic Diagram of Extraction Framework

3. EXPERIMENTAL RESULT SET ANALYSIS

Our proposed framework is developed and carried out with detailed experiment using Java 2 Standard Development Kit, Version 1.6 and Windows environment. We've tested proposed framework with several color images and a set of excellent result set has been found. In this section we've highlighted the following points:

- Persistency of quality after embedding the watermark / invisibility to HVS.
- Recognizable watermark extraction after high JPEG compression.

3.1 Persistency of quality after embedding the watermark / invisibility to HVS

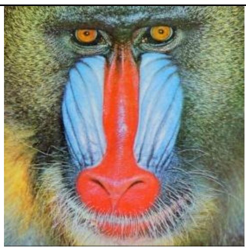
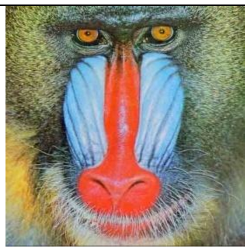
Proposed framework has been tested on more than thousand samples and it is ensured that the watermarked images successfully carried the persistency of their quality. Four of our tested result sets are in Table 1. We've tested our framework with different quality factors for JPEG compression— 50 to 30 whereas the quality factor can range between 1 to 100, where 1 gives the poorest quality image and highest compression and 100 gives the best image quality and lowest compression [8].

It is clear that after embedding the grayscale watermark to color host images, the watermark is perceptually invisible to the HVS and 5th column of Table.1 is provided in this regard. The PSNR is most commonly used as a measure of quality of watermarked image. It is most easily defined via the root mean squared error (RMSE) as described in Equation8 [9].

$$PSNR = 20\, log_{10}\left(\frac{MAX}{RMSE}\right) \tag{8}$$

Table 1. PSNR Analysis

Host Image	Watermark	Watermarked Image	Quality Factor for JPEG Compression	PSNR (dB)
			50	50.29
			40	44.17
			30	37.89

			50	57.29
			40	51.48
			30	43.37
			50	51.97
			40	46.45
			30	38.23

If a pixel in the host image is defined as Y (i, j) and that in the watermarked image is defined as y (i, j), then the root mean square error (RMSE) of the watermarked image is computed with Equation9 [10].

$$RMSE = \sqrt{\frac{\sum_{i=0}^{M-1}\sum_{j=0}^{N-1}[Y_{(i,j)} - y_{(i,j)}]^2}{M \times N}} \quad (9)$$

Some more result set regarding PSNR values are depicted at Table 2 reported below.

Table 2. PSNR with JPEG Compression

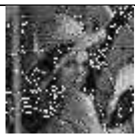
Host Image Name	**Resolution (Pixel)**	**Watermark**	**PSNR with respective quality factor for JPEG compression**		
			50	**40**	**30**
Girl.png	512x512	Logo1.jpg	50.29	44.17	37.89
Bride.jpg	512x512	Logo2.jpg	56.12	48.23	40.46
Baboon.jpg	512x512	Logolenna.jpg	57.38	49.33	43.27
Pakhi.png	1024x1024	Logo3.jpg	63.22	54.87	49.61
Tower.jpg	1024x1024	Logobaboon.jpg	66.11	55.12	50.58
Vegitable.jpg	1024x1024	Logogirl.jpg	67.25	57.34	51.98
Parrot.jpg	1024x1024	Logo4.jpg	68.19	61.41	53.30

3.2 Recognizable watermark extraction after high JPEG compression.

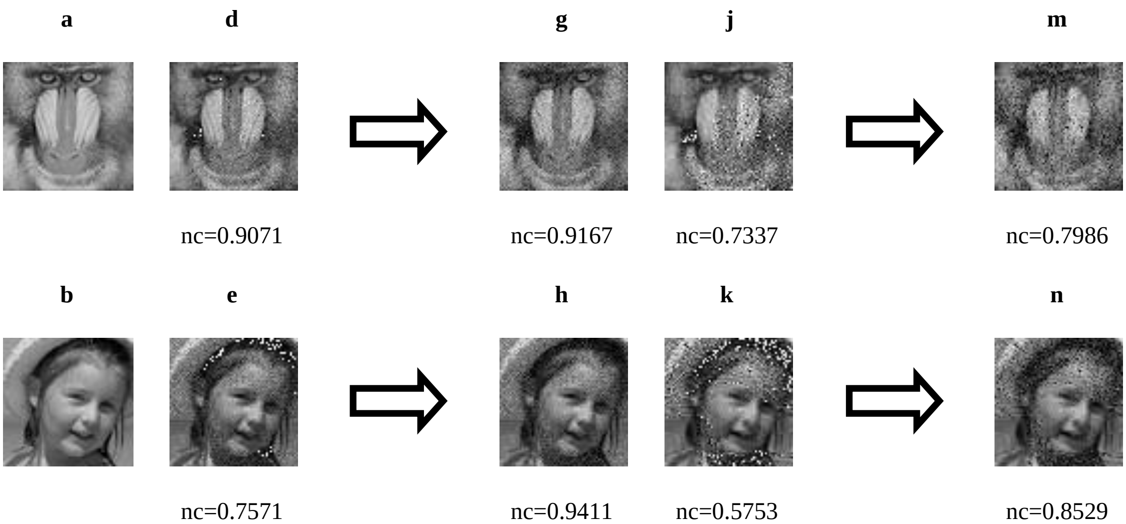
Blind extraction is used at the extraction end i.e., neither the host image nor the watermark is needed at the time of extraction. After performing JPEG compression at various rates, the watermark is easily recognizable in HVS. The quantitive similarity measurement between the referenced watermark and extracted watermark is computed by normalized correlation (nc). 5th column of Table 3 shows the nc values for different watermarks and conforms the excellency of proposed framework. The nc calculation is done with Equation10 [11].

$$nc = \frac{\sum\sum(I_w[i][j]*I_o[i][j])}{\sqrt{\sum\sum(I_w[i][j]*I_o[i][j])^2}} \qquad (10)$$

Table 3. nc Value Analysis Extracted Watermark

Reference Watermark	Extracted Watermark	Quality Factor for JPEG Compression	nc
		50	0.9071
		30	0.7337
		50	0.7479
		30	0.5652

As we can see after high JPEG compession the retrieved watermarks are suffered from salt and pepper noise. So we've removed this noise to have a better result. The method can be found at [12]. Fig.6 shows how excellently retrieved watermarks are reconstructed by removing the noise. The nc values are also greatly improved with the reduction of noise.



c **f** **i** **l** **o**

nc=0.7479 nc=0.9276 nc=0.5652 nc=0.8572

Fig.6 nc Analysis after removal of salt and pepper noise of retrieved watermarks after JPEG compression with different quality factors. [a-c]— reference watermarks; [d-f]— retrieved watermarks after JPEG compression with quality factor 50; [g-i]— reconstructed watermarks after noise reduction; [j-l]— retrieved watermarks after JPEG compression with quality factor 30; [m-o]— reconstructed watermarks after noise reduction.

4. CONCLUSION

In this article a novel DCT based framework for invisible color image watermarking is proposed. It is proven that the watermarking framework is highly robust against high JPEG compression (i.e., low in value). The quality of the watermarked image is well preserved and invisibility of grayscale watermark is guaranteed. In this regard we got a brilliant set of PSNR values. Security issue is greatly preserved with secret key. The key is dynamic in length, i.e., depending on the size of the watermark the length of the key is calculated and an algorithm is also proposed to choose the embedding blocks with the said secret key. The watermark extraction does not involve the host image or watermark i.e., blind extraction framework is ensured. Even after performing high JPEG compression, we've got a good set of nc values. However higher degree of JPEG compression will lead to some salt and pepper noise in the extracted watermark but that can also be removed and nc can be increased greatly through the adopted noise reduction algorithm. As a future scope more different types of attack can be taken into consideration.

5. REFERENCES

[1] Ingemar J. Cox, Matthew L. Miller, Jeffrey A. Bloom, Jessica Fridrich, Ton Kalker, "Digital Watermarking and Steganography", 2nd Edition, Morgan Kaufmann Publishers, 2008. (ISBN: 978-0-12-372585-1).

[2] Soumik Das, Pradosh Bandyopadhyay, Shauvik Paul, Arindam Sinha Ray, Dr. Monalisa Banerjee— "Pioneering the Technique for Invisible Image Watermarking on Color Image", International Journal of Recent Trends in Engineering (IJRTE), May Issue, 2009, (ISSN: 1787-9617).

[3] Soumik Das, Pradosh Bandyopadhyay, Shauvik Paul, Arindam Sinha Ray, Dr. Monalisa Banerjee— "A New Introduction towards Invisible Image Watermarking on Color Image", IEEE International Advance Computing Conference (IACC'09), Thapar University, India, March, 2009. (ISBN: 978-1-4244-2927-1).

[4] Pradosh Bandyopadhyay, Soumik Das, Shauvik Paul, Prof. Atal Chaudhuri, Dr. Monalisa Banerjee— "A Dynamic Watermarking Scheme for Color Image Authentication", IEEE International Conference on Advances in Recent Technologies in Communication and Computing (ARTCom-2009), Kerala, India, October 2009. (ISBN: 978-0-7695-3845-7).

[5] Soumik Das, Pradosh Bandyopadhyay, Dr. Monalisa Banerjee, Prof. Atal Chaudhuri— "A Chip-Based Watermarking Framework for Color Image Authentication for Secured `Communication", International Conference in Computing, Communication and Control-2011 (ICAC3-11), Mumbai, India, January, 2011. Published by Springer in Communications in Computer and Information Science, 1, Volume 125, Advances in Computing, Communication and Control, Part 2, Pages 349-354.

[6] Pradosh Bandyopadhyay, Soumik Das, Dr. Monalisa Banerjee, Prof. Atal Chaudhuri— "Fragile Color Image Watermarking Framework Focusing A New Invisible Domain", International Conference on Communication, Computing & Security (ICCCS2011), NIT Rourkela, India, February, 2011, Published by ACM (ISBN-978-1-4503-0464-1).

[7] Shinfeng D. Lin, Shih-Chieh Shie, and Jim Yi Guo— "Improving the Robustness of DCT-Based Image Watermarking Against JPEG Compression", Jan 2010 Elsevier, Journal of Computer Standard and Interfaces Volume 32, Issues 1-2, Pages 54-60.

[8] K. R. Rao, Ping Wip— "Discrete Cosine Transform: algorithms, advantages, applications", 1990, Academic Press.

[9] Soumik Das, Pradosh Bandyopadhyay, Shauvik Paul, Prof. Atal Chaudhuri, Dr. Monalisa Banerjee— "An Invisible Color Watermarking Framework for Uncompressed Video Authentication", International Journal of Computer Applications (IJCA), and also available at Google Scholar and DBLP, Volume1, Number 11, March Issue, 2010, (ISBN: 978-93-80746-10-4).

[10] Soumik Das, Pradosh Bandyopadhyay, Dr. Monalisa Banerjee, Prof. Atal Chaudhuri— "A Chip-Based Watermarking Framework for Color Image Authentication", International Conference on Communication, Computing & Security (ICCCS2011), NIT Rourkela, India, February, 2011, Published by ACM (ISBN-978-1-4503-0464-1).

[11] Pradosh Bandyopadhyay, Soumik Das, Prof. Atal Chaudhuri, Dr. Monalisa Banerjee— "A New Invisible Color Image Watermarking Framework through Alpha Channel", IEEE International Conference on Advances in Engineering, Science and Management (IEEE-ICAESM 2012), India, March, 2012. (ISBN: 978-1-4673-0213-5).

[12] Raymond H. Chan, Chung-Wa Ho, and Mila Nikolova— "Salt-and-Pepper Noise Removal by Median-Type Noise Detectors and Detail-Preserving Regularization", 2005 IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 14, NO. 10, Page(s): 1479- 1485. Digital Object Identifier 10.1109/TIP.2005.852196.