Journal of Engineering and Sustainable Development

Vol. 22, No.06, November 2018
ISSN 2520-0917
https://doi.org/10.31272/jeasd.2018.6.1

A DISCRETE COSINE TRANSFORM BASED WATERMARKING SCHEME FOR COLOR IMAGE USING YCbCr SPACE

Adel Jalal Yousif

Assistant Lecturer, Electronic computer center, University of Diyala, Diyala, Iraq.

Abstract: In this paper, a non-blind color image watermarking scheme is proposed based on discrete cosine transform and YCbCr color space. The host image is converted from RGB space into YCbCr space and the Y component image is employed for embedding a binary watermark image. The strength of the watermark is controlled by a robustness factor to obtain an acceptable trade-off between robustness and imperceptibility. The performance of the proposed scheme is evaluated in term of Peak signal to noise ratio (PSNR) and normalized correlation (NC). The experimental results show that the proposed scheme is robust under various attacks like JPEG compression, filtering and noise addition.

Keywords: *Discrete Cosine Transform (DCT), Digital watermarking, YCbCr color space, Transform domain.*

نظام اخفاء العلامة المائية للصور الملونة بالاعتماد على تحويلات جيب التمام المتقطعة ونظام الألوان YCbCr

الخلاصة: في هذا المقال تم تقديم نظام لا خفاء صورة تمثل علامة مائية ضمن صورة اخرى بالاعتماد على تحويلات جيب التمام المتقطعة (DCT) وعلى نظام الالوان (YCbCr). تضمن العلامة المائية في المركبة (Y) بعد تحويل نظام الالوان للصورة المضيفة من (RGB) الى نظام (YCbCr). يتم التحكم بقوة تأثير العلامة المائية على الصورة المضيفة عن طريق معامل القوة لإيجاد موازنة بين قوة نظام الاخفاء من جهة والحفاظ على جودة الصورة المضيفة من جهة اخرى. تم استخدام مقياس ذروة الاشارة إلى نسبة الضوضاء (PSNR) ومقياس علاقة الترابط (normalized correlation) لتقييم اداء النظام المقترح. بينت التجارب العملية مقاومة النظام المقترح للعديد من عمليات معالجة الصور الرقمية مثل عملية ضغط الصور باستخدام تقنية (JPEG) وعمليات الترشيح وعمليات اضافة الضوضاء.

1. Introduction

With the quick advancement of computer and network technology, computerized information can now be transmitted through internet quick and simple. Since the digitized information could be unlawfully copied and effortlessly altered, the enforcement of multimedia copyright protection became an essential issue. One of the effective approach to take care of this issue is digital watermarking for copyright

**Corresponding Author* ajalal289@gmail.com

protection. Digital watermarking is the process of embedding information (watermark) into a multimedia content (audio, image or video) in some way the embedded information can be identified or extracted later for the purpose of copy right protection, authentication, ownership verification, broadcast monitoring, etc. The quality of watermarking scheme is determined by four major factors including robustness, imperceptibility, security and capacity [1]. A good watermarking scheme should be robust against a variety of possible attacks like scaling, compression, cropping, adding noise, filtering and others without distortion the host image quality (imperceptibility). Thus, there is a need to improve watermarking algorithms that can offer good balance between imperceptibility and robustness. Generally, image watermarking methods can be implemented either in spatial domain or in frequency domain. In spatial domain, the inserting of watermark is done by directly modifying the pixels value in such a way that remaining the watermark is invisible. Least significant bit (LSB) is one of the most commonly used algorithms in spatial domain. In transform domain, the pixel value is substituted with the transform coefficient and the embedding process is done by altering the transform domain coefficients. Most commonly used transform domain algorithms are discrete cosine transform (DCT) and discrete wavelet transform (DWT) [2]. The transform domain techniques are more effective than the spatial domain for achieving the robustness and imperceptibility, as clarified by various surveys [3-5]. Hence, the transform domain is utilized for the proposed scheme.

2. Discrete Cosine Transform

The Discrete Cosine Transform (DCT) is an orthogonal transformation that is very widely used linear transform in digital signal processing. DCT decomposes a signal into a series of symmetric cosine functions therefore the obtained matrix is determined by the horizontal, vertical, and diagonal frequencies. The key feature of the DCT decomposition is that the vast majority of the signal information tends to be collected in a few low-frequency components of the DCT and the part that the eye of human is least sensitive to is neglected. Thus, it is very widely used in image compression and is widely accepted in the multimedia standards. The two dimensional discrete cosine transform and inverse discrete cosine transform (IDCT) for an image $M \times N$ is described by the following equations [6]:

$$F(u,v) = c(u)c(v)\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)\cos\frac{\pi(2x+1)}{2M}\cos\frac{\pi(2y+1)}{2N} \qquad (1)$$

$$c(u) = \begin{cases}\sqrt{1/M} & u = 0\\ \sqrt{2/M} & u = 1,2,\dots,M\end{cases} \qquad (2)$$

$$c(v) = \begin{cases}\sqrt{1/N} & v = 0\\ \sqrt{2/N} & v = 1,2,\dots,N\end{cases} \qquad (3)$$

$$f(x,y) = \sum_{u=0}^{M-1}\sum_{v=0}^{N-1} c(u)c(v)F(u,v)\cos\frac{\pi(2x+1)u}{2M}\cos\frac{\pi(2y+1)v}{2N} \qquad (4)$$

3. YCbCr Color Space

The digital image watermarking is often accomplished in gray scale but color image watermarking can be employed to improve the performance of the watermarking scheme. Color spaces are three-dimensional systems used to represent specific organization of colors depending on three coordinates that describe color. The RGB space is the basic one, that can be transformed into other color spaces. YCbCr color space represents each color with three components. The Y component describes the light intensity (luminance). The Cb and Cr components are describe the color information. Thus, YCbCr color space separates brightness and chroma. In the proposed scheme, the Y component is used for watermark embedding process to satisfy robustness quality measures. The RGB to YCbCr conversion is defined as [7]:

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.16875 & -0.33126 & 0.500 \\ 0.500 & -0.41869 & -0.08131 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \qquad (5)$$

The inverse conversion from YCbCr to RGB space is defined as:

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1.0 & 0.0 & 1.402 \\ 1.0 & -0.34413 & -0.71414 \\ 1.0 & 1.772 & 0.0 \end{bmatrix} \begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} \qquad (6)$$

4. Proposed Scheme

In the proposed scheme, a binary watermark image is embedded in color image by using DCT and YCbCr color space. The proposed scheme employs the properties of DCT and YCbCr space to increase the robustness and imperceptibility of the watermarking algorithm by using the most significant coefficients of the whole Y component image. The watermark embedding and extraction algorithms are clarified in the following sections and illustrated in Figs. 1 and 2 respectively.

4.1. Watermark Embedding Algorithm

The watermark embedding algorithm is described by the following steps:

Step 1: Read the original color image I with the size of $M \times N$ and the binary watermark image W with the size of $P \times Q$.

Step 2: Convert the RGB space of the original image into YCbCr space and select the Y component.

Step 3: The $M \times N$ DCT for the $M \times N$ (Y component) is calculated to produce y matrix.

Step 4: Select the (n) maximum DCT coefficients which are denoted by $T = \{t_1, t_2, \ldots . t_n\}$ (excluding the DC term located in (0,0) of the DCT matrix y),

where: $n = P \times Q$, $T \in y$.

Step 5: Take an element $W(i,j)$ of the watermark image, embedded in the selected coefficients of T according to the following formula:

$$t'_k = t_k * (1 + b * \alpha) \qquad (7)$$

Where α is the robustness factor, the value of α is discussed in section 5.

$k = 1,2, \ldots , n$.

$$b = \begin{cases} -1, & W(i,j) = 1 \\ 1, & W(i,j) = 0 \end{cases} \text{, where: } 1 \le i \le P,\ 1 \le j \le Q \ .$$

Step 6: Apply the IDCT to the altered Y component y' and combine the YCbCr space components

Step 7: Obtain the watermarked image I′ by switching the color space from YCbCr to RGB.

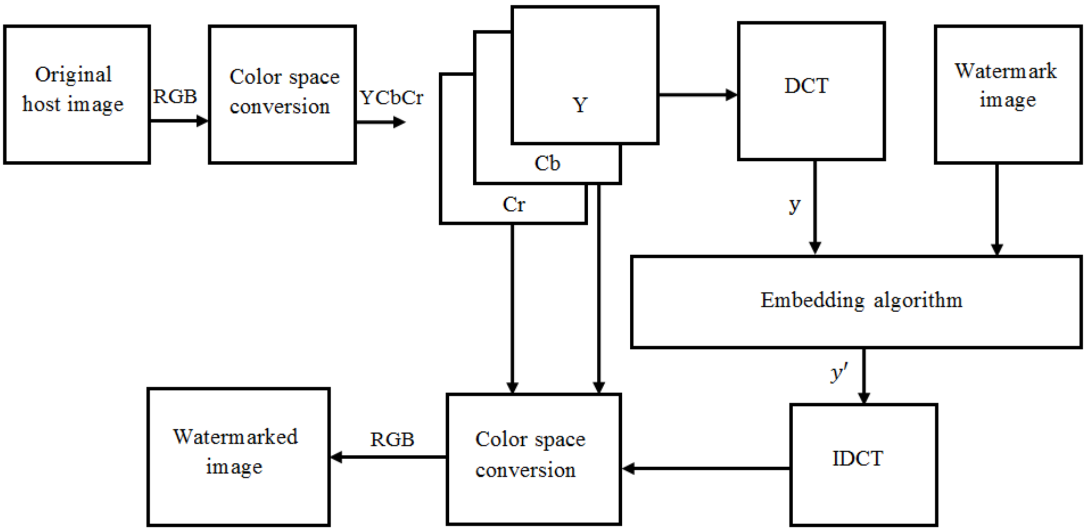


Figure 1. Watermark embedding algorithm.

4.2. Watermark Extraction Algorithm

The watermark extraction algorithm is described by the following steps:

Step 1: Read the original image I and watermarked image I′ .

Step 2: Convert the RGB space of the original and watermarked images into YCbCr space and select the Y components .

Step 3: Apply the DCT to the whole Y components to obtain y and y'.

Step 4: Obtain the (n) maximum DCT coefficients $T' = \{t'_1, t'_2, \ldots, t'_n\}$ of y' according to the DCT coefficients $T = \{t_1, t_2, \ldots, t_n\}$, where the DCT coefficients of T and T' have the same positions in the DCT matrices of y and y' respectively.

Step5: let $\sigma_k = (t'_k / t_k) - 1$ (8)

where $= 1,2, \ldots, n$.

Step6: Extract the watermark image W^e with the size of $P \times Q$ by making the judgment of σ_k :

$$W^e(i.j) = \begin{cases} 0, & \sigma_k > 0 \\ 1, & \sigma_k \leq 0 \end{cases} \qquad (9)$$

where: $1 \leq i \leq P$, $1 \leq j \leq Q$.

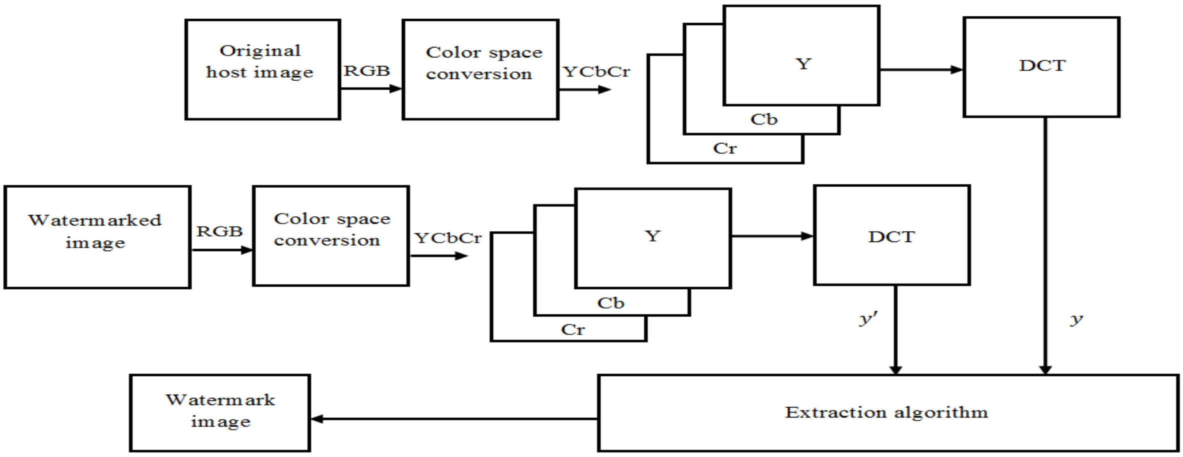


Figure 2. Watermark extraction algorithm.

5. Experimental Results and Simulation

The quality of the digital watermarking techniques is commonly estimated by the robustness of the watermark to the common signal processing operations of the watermarked image and the imperceptibility of the embedded watermark to human observers. In this paper, the proposed scheme is evaluated for both properties of robustness and imperceptibility by using Matlab platform. Three 512×512 well-known color images: Airplane, Peppers, and Baboon shown in Fig. 3(a-c) are used as the host images for embedding a 30×40 binary watermark image, shown in Fig. 3(d).

(a) (b) (c) (d)

Figure 3. Test images used for evalution, (a) Airplane, (b) Peppers, (c) Babbon and (d) watermark logo.

Different attacks including noise addition, filtering and JPEG compression are used to test the robustness of the watermark. For evaluating the robustness of the proposed scheme, the Normalized Correlation (NC) is used to measure the similarity degree between the original watermark image (W) and the extracted watermark W^e. When the NC value is 1, means that the original and the extracted watermark is precisely similar. NC formula for two images is given by [8]:

$$\text{NC} = \frac{\sum_{i=1}^{P}\sum_{j=1}^{Q}(W(i,j) \times W^e(i,j))}{\sum_{i=1}^{P}\sum_{j=1}^{Q}(W(i,j))^2} \qquad (10)$$

For evaluating the imperceptibility of the embedded watermark, the peak signal to noise ratio (PSNR) is used to measure the quality of the watermarked image.

The higher the PSNR value the lower the perception of the watermark in the host image. PSNR is given by [8]:

$$\text{PSNR} = 10\log_{10}\left(\frac{255^2}{\text{MSE}}\right) \qquad (11)$$

Where the MSE is the mean square error and defined as:

$$\text{MSE} = \frac{\sum_{x=1}^{M}\sum_{y=1}^{N}(I(x,y) - I'(x,y))}{M \times N} \qquad (12)$$

For achieving an acceptable tradeoff between robustness and imperceptibility, the strength of the proposed watermarking scheme is controlled by a factor (α). In the other word, the value of α determine the strength of the watermark on the host image. theoretically, for high values of α, the robustness of the watermarked image increases while its quality decreases and vice versa. Fig. 4 shows this effect visually according to human observer and Tables 1 and 2 show this effect in quantitative manner.

(a) Original (b) Watermarked α = 0.05 PSNR = 39.64 (c) Watermarked α = 0.1 PSNR = 33.87 (d) Watermarked α = 0.2 PSNR = 27.92

Figure 4. The effect of α value on the watermarked image quality.

Table 1. Performance comparison1 with varying values of α (from 0.01 to 0.2).

Images	α	No attack		JPEG (QF=10) compression		JPEG(QF=20) compression		JPEG (QF=30) compression		Average filtering (3×3)		Median filtering (3×3)	
		PSNR	NC	PSNR	NC	PSNR	NC	PSNR	NC	PSNR	NC	PSNR	NC
Airplan	0.01	50.24	0.9305	26.95	0.5695	29.02	0.6379	30.21	0.6659	29.88	0.9215	34.23	0.6401
	0.05	41.17	1.0000	26.85	0.7769	28.82	0.9092	29.93	0.9596	29.58	0.9507	33.54	0.9540
	0.1	35.50	1.0000	26.49	0.9137	28.19	0.9843	29.13	1.0000	28.82	0.9574	31.90	0.9966
	0.15	32.04	1.0000	25.86	0.9585	27.33	1.0000	28.07	1.0000	27.80	0.9596	30.07	1.0000
	0.2	29.57	1.0000	25.17	0.9944	26.34	1.0000	26.90	1.0000	26.69	0.9652	28.36	1.0000
Pepper	0.01	49.32	0.9103	28.79	0.6491	32.25	0.6704	34.29	0.7130	35.28	0.9002	41.36	0.8913
	0.05	39.64	1.0000	28.48	0.8184	31.56	0.9182	33.25	0.9753	33.93	0.9765	37.70	1.0000
	0.1	33.87	1.0000	27.63	0.9305	29.99	0.9854	31.09	1.0000	31.44	0.9821	33.27	1.0000
	0.15	30.40	1.0000	26.56	0.9697	28.23	0.9989	28.94	1.0000	29.11	0.9854	30.12	1.0000
	0.2	27.92	1.0000	25.35	0.9922	26.53	1.0000	27.03	1.0000	27.14	0.9888	27.76	1.0000
Baboon	0.01	51.20	0.8475	21.70	0.5404	23.16	0.5751	23.99	0.6166	22.77	0.8655	23.18	0.4742
	0.05	42.90	1.0000	21.67	0.6996	23.11	0.8565	23.94	0.9193	22.73	0.9563	23.18	0.7287
	0.1	37.41	1.0000	21.59	0.8397	23.00	0.9664	23.80	0.9966	22.63	0.9630	23.10	0.8935
	0.15	34.00	1.0000	21.46	0.9260	22.81	0.9944	23.57	1.0000	22.46	0.9686	22.95	0.9507
	0.2	31.54	1.0000	21.27	0.9709	22.57	1.0000	23.27	1.0000	22.23	0.9753	22.73	0.9742

Table 2. Performance comparison2 with varying values of α (from 0.01 to 0.2).

Images	α	Salt and pepper noise (1%)		Gaussian noise (μ=0,υ=0.005)		Speckle noise (1%)		Cropping 10 rows		Histogram stretching	
		PSNR	NC	PSNR	NC	PSNR	NC	PSNR	NC	PSNR	NC
airplane	0.01	24.85	0.7399	23.04	0.6570	22.68	0.6054	20.12	0.6155	28.15	0.2374
	0.05	24.74	0.9630	22.97	0.8957	22.64	0.9002	20.09	0.8262	26.06	0.6587
	0.1	24.68	0.9922	22.81	0.9854	22.50	0.9742	20.00	0.9013	25.18	0.8610
	0.15	24.09	1.0000	22.52	1.0000	22.25	0.9978	19.85	0.9294	24.61	1.0000
	0.2	23.61	1.0000	22.18	1.0000	21.93	1.0000	19.66	0.9428	24.00	1.0000
peppers	0.01	24.82	0.7780	23.17	0.7578	26.59	0.6614	22.33	0.5942	43.68	0.4283
	0.05	24.76	0.9608	23.05	0.9271	26.40	0.9462	22.25	0.7186	38.07	0.8689
	0.1	24.48	0.9955	22.76	0.9922	25.83	0.9955	22.03	0.8442	33.00	0.9978
	0.15	23.90	1.0000	22.35	0.9966	25.03	1.0000	21.68	0.9047	30.16	0.9989
	0.2	23.21	1.0000	21.85	1.0000	24.12	1.0000	21.24	0.9439	27.86	1.0000
baboon	0.01	25.24	0.6827	23.08	0.5919	25.52	0.5381	23.40	0.5886	42.81	0.3146
	0.05	25.07	0.9058	23.03	0.8453	25.46	0.8733	23.36	0.7567	38.02	0.8049
	0.1	25.01	0.9899	22.92	0.9675	25.28	0.9854	23.24	0.8487	34.72	1.0000
	0.15	24.82	1.0000	22.74	0.9955	24.96	0.9966	23.04	0.8890	32.78	1.0000
	0.2	24.39	1.0000	22.49	1.0000	24.58	1.0000	22.78	0.9159	30.90	1.0000

Tables 1, 2 and Fig. 4 show that the value of (α) should be chosen carefully to obtain a good balance between the robustness of extracted watermark and the imperceptibility of watermarked image. In these tables, the PSNR is calculated by obtaining the average PSNR value for the red, green and blue channels [9]. Fig. 5 and 6 show the watermarked peppers images and the detected watermark logo images after subjecting the watermarked image to different known attacks with the corresponding values of PSNR and NC.

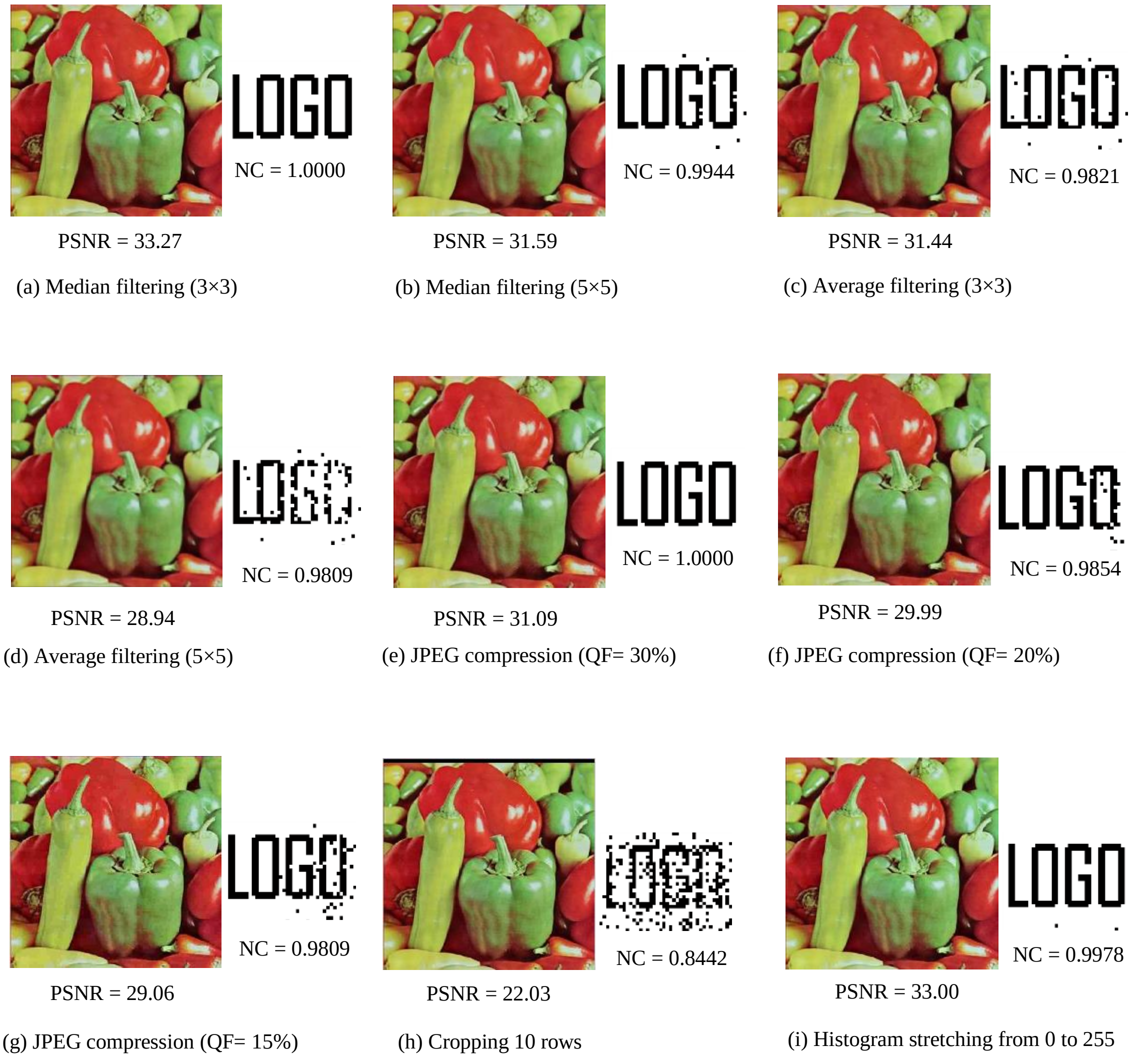
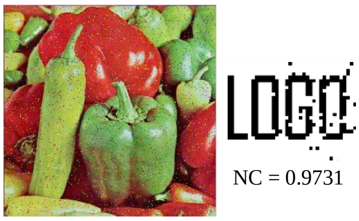
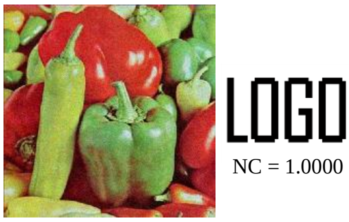


Figure 5: Watermarked Peppers images subjected to different image processing operations attacks and the extracted watermark logo image of each one with $\alpha = 0.1$.

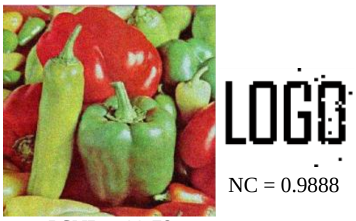
PSNR = 24.45
(a) Salt and pepper noise (1%)



PSNR = 20.00
(b) Salt and pepper noise (3%)

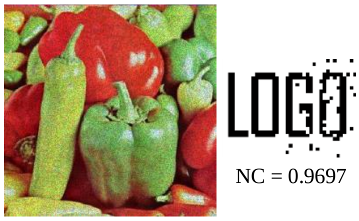


PSNR = 26.21
(c) Gaussian noise (μ=0,v=0.002)



PSNR = 22.78
(d) Gaussian noise (μ=0,v=0.005)

PSNR = 25.84
(e) Speckle noise (1%)



PSNR = 21.68
(f) Speckle noise (3%)

Figure 6: Watermarked Peppers images subjected to different noise attacks and the extracted watermark logo image of each one with = 0.1 .

Tables 3-5 present performance analysis and comparison based on PSNR and NC for the tested images that exposed to different attacks including (JPEG compression with different quality factor (QF), Gaussian noise, salt and peppers noise, speckle noise, average filtering, median filtering, cropping and histogram stretching).

Table 3. Performance evaluation and comparison for Airplane image.

Attacks	Parameter	Proposed scheme for α = 0.1		Proposed scheme for α = 0.05		Scheme in [10]	
		PSNR	NC	PSNR	NC	PSNR	NC
No attack	-	35.50	1.0000	41.17	1.0000	45.23	0.9989
JPEG compression	(QF=45)	29.73	1.0000	30.69	0.9910	30.93	0.7130
JPEG compression	(QF=35)	29.38	1.0000	30.24	0.9776	30.48	0.4540
JPEG compression	(QF=25)	28.73	0.9966	29.46	0.9540	29.70	0.3038
JPEG compression	(QF=15)	27.56	0.9675	28.05	0.8744	28.20	0.2242
JPEG compression	(QF= 5)	24.26	0.8016	24.44	0.7253	24.49	0.0751
Salt and peppers	(density = 0.005)	27.21	1.0000	27.82	0.9865	27.86	0.9563
Salt and peppers	(density = 0.02)	21.68	0.9899	21.81	0.9406	21.83	0.8150
Salt and peppers	(density = 0.05)	17.82	0.9765	17.87	0.9159	17.94	0.7309
Gaussian noise	(μ=0,v=0.002)	26.41	1.0000	26.82	0.9753	26.92	0.9070
Gaussian noise	(μ=0,v=0.004)	23.70	0.9933	23.91	0.9182	23.97	0.8397
Gaussian noise	(μ=0,v=0.006)	22.07	0.9798	22.22	0.8901	22.25	0.8016
Speckle noise	(density = 0.005)	25.28	0.9989	25.59	0.9507	25.65	0.8610
Speckle noise	(density = 0.02)	19.69	0.9574	19.76	0.8397	19.80	0.7108
Speckle noise	(density = 0.05)	16.29	0.9249	16.36	0.8374	16.40	0.6357
Average filtering	(3×3)	28.82	0.9574	29.58	0.9507	29.89	0.9518
Average filtering	(5×5)	25.74	0.9574	26.13	0.9518	26.29	0.2332
Median filtering	(3×3)	31.90	0.9966	33.54	0.9540	34.19	0.9563
Median filtering	(5×5)	28.88	0.9563	29.61	0.8363	29.87	0.4417
Cropping	10 rows	20.00	0.9013	20.09	0.8262	20.11	0.9552
Histogram stretching	From 0 to 255	25.18	0.8610	26.06	0.6587	32.36	0.9989

Table 4. Performance evaluation and comparison for Peppers image.

Attacks	Parameter	Proposed scheme for $\alpha = 0.1$		Proposed scheme for $\alpha = 0.05$		Scheme in [10]	
		PSNR	NC	PSNR	NC	PSNR	NC
No attack	-	33.87	1.0000	39.64	1.0000	46.12	1.0000
JPEG compression	(QF=45)	31.90	1.0000	34.67	0.9922	36.01	0.6659
JPEG compression	(QF=35)	31.45	0.9989	33.82	0.9832	34.98	0.5448
JPEG compression	(QF=25)	30.59	0.9955	32.47	0.9395	33.33	0.4686
JPEG compression	(QF=15)	29.06	0.9664	30.32	0.8599	30.86	0.4002
JPEG compression	(QF= 5)	24.24	0.7859	0.8599	0.6872	24.81	0.3666
Salt and peppers	(density = 0.005)	26.90	1.0000	27.55	0.9742	27.88	0.94
Salt and peppers	(density = 0.02)	21.74	0.9865	21.91	0.9204	21.94	0.8206
Salt and peppers	(density = 0.05)	17.88	0.9641	17.94	0.8913	17.97	0.7242
Gaussian noise	(μ=0,v=0.002)	26.22	1.0000	26.84	0.9608	27.06	0.8767
Gaussian noise	(μ=0,v=0.004)	23.63	0.9922	23.98	0.9249	24.11	0.8229
Gaussian noise	(μ=0,v=0.006)	22.06	0.9832	22.30	0.9070	22.41	0.7870
Speckle noise	(density = 0.005)	28.14	1.0000	29.17	0.9664	29.49	0.8857
Speckle noise	(density = 0.02)	23.25	0.9787	23.54	0.8890	23.60	0.7534
Speckle noise	(density = 0.05)	19.69	0.9496	19.81	0.8173	19.83	0.6379
Average filtering	(3×3)	31.44	0.9821	33.93	0.9765	35.35	0.9619
Average filtering	(5×5)	28.94	0.9809	30.21	0.9675	30.83	0.2265
Median filtering	(3×3)	33.27	1.0000	37.70	1.0000	41.28	0.9765
Median filtering	(5×5)	31.59	0.9944	34.07	0.9731	35.37	0.5404
Cropping	10 rows	22.03	0.8442	22.25	0.7186	22.32	0.9540
Histogram stretching	From 0 to 255	33.00	0.9978	38.07	0.8689	43.19	1.0000

Table 5. Performance evaluation and comparison for Baboon image.

Attacks	Parameter	Proposed scheme for $\alpha = 0.1$		Proposed scheme for $\alpha = 0.05$		Scheme in [10]	
		PSNR	NC	PSNR	NC	PSNR	NC
No attack	-	37.41	1.0000	42.90	1.0000	40.61	0.9518
JPEG compression	(QF=45)	24.61	1.0000	24.78	0.9798	24.74	0.9193
JPEG compression	(QF=35)	24.11	0.9966	24.26	0.9496	24.22	0.8318
JPEG compression	(QF=25)	23.44	0.9966	23.56	0.9182	23.54	0.7298
JPEG compression	(QF=15)	22.43	0.9339	22.54	0.7982	22.53	0.6166
JPEG compression	(QF= 5)	19.86	0.7433	19.93	0.6693	19.95	0.3296
Salt and peppers	(density = 0.005)	27.85	1.0000	28.09	0.9641	27.95	0.9204
Salt and peppers	(density = 0.02)	22.13	0.9720	22.21	0.8868	22.18	0.8475
Salt and peppers	(density = 0.05)	18.20	0.9350	18.27	0.8599	18.18	0.7444
Gaussian noise	(μ=0,v=0.002)	26.63	0.9944	26.91	0.9148	26.85	0.8812
Gaussian noise	(μ=0,v=0.004)	23.86	0.9720	23.98	0.8812	23.95	0.8442
Gaussian noise	(μ=0,v=0.006)	22.18	0.9428	22.27	0.8296	22.26	0.8061
Speckle noise	(density = 0.005)	27.96	1.0000	28.32	0.9361	28.21	0.9058
Speckle noise	(density = 0.02)	22.48	0.9395	22.59	0.8150	22.57	0.8184
Speckle noise	(density = 0.05)	18.76	0.8677	18.81	0.7287	18.81	0.7422
Average filtering	(3×3)	22.63	0.9630	22.73	0.9563	22.75	0.7780
Average filtering	(5×5)	20.60	0.9596	20.66	0.9552	20.69	0.3946
Median filtering	(3×3)	23.10	0.8935	23.18	0.7287	23.15	0.7545
Median filtering	(5×5)	20.84	0.8072	20.87	0.6827	20.86	0.4283
Cropping	10 rows	23.24	0.8487	23.36	0.7567	23.33	0.9036
Histogram stretching	From 0 to 255	34.72	1.0000	38.02	0.8049	39.00	0.9518

6. Conclusions

This paper offers a non-blind watermarking method for color images based on DCT and color space conversion. Since the YCbCr color space separates brightness and chroma, the use of Y component for watermark embedding is led to improvement in results. The watermarked image is exposed to different attacks (JPEG compression, Gaussian noise, salt and peppers noise, speckle noise, median filtering, average filtering, cropping and histogram stretching) for performance evaluating. Figs. 4(b) and 4(c) show that the proposed scheme attains the imperceptibility of the embedded watermark against the human observer . According to Tables 3-5, it is obvious that the proposed scheme achieves both the robustness and the imperceptibility under different attacks.

7. References

1. V. Madhu Viswanatham, Galma Santosh Reddy and Potluri Jagadeesh. (2012). *"A hybrid digital watermarking algorithm for color images based on DWT and DCT"*, 10th Tome 2nd Fasc, Annals- Computer Science Series.
2. Sanpreet Singh, Savina Bansal and Sukhjinder Singh. (2015). *"Robust and Secure Image Watermarking using DWT-SVD and Chaotic Map "* , International Journal of Advanced Research in Computer and Communication Engineering Vol. 4, Issue 9.
3. Preeti Parashar and Rajeev Kumar Singh. (2014). *"A survey: Digital Image Watermarking Techniques"*, International Journal of signal processing, image processing and pattern recognition, Vol. 7, No. 6, pp. 111-124.
4. Seema Malshe, Hitesh Gupta and Saurabh Mandloi. (2012). *"Survey of Digital Image Watermarking Techniques to achieve Robustness"*, International journal of computer applications, Vol. 45. No.13.
5. Meenu Singh, Abhishek Singhal and Ankur Chaudhary. (2013). *"Digital Image Waermarking Techniques: A Survey"*, International journal of computer science and telecommunications, Vol.4, Issue 6.
6. Ying Yang and Haiping Li. (2015). *" The Application of DCT Algorithm in Digital Watermarking by Matlab and Simulation"*, 7th International Conference on Modeling, Identification and Control (ICMIC).
7. Mehdi Khalili and David Asatryan. (2012). *"Improved DWT Based Watermarking Using JPEG-YCbCr"*, CSIT Conference, Yerevan, Armenia.
8. Baisa L. Gunjal and Suresh N.Mali. (2011). *"Secured Color Image Watermarking Technique in DWT-DCT Domain"*, International journal of computer science, engineering and information technology, Vol. 1, No. 3.

9. Thien Huynh, Oresti Banos, Sungyoung Lee, Yongik Yoon and Thuong Le-Tien. (2016). *"Improving digital image watermarking by means of optimal channel selection"*, International Journal of Expert Systems with Applications, Vol. 62, pp177-189.
10. Xiangguang Xiong (2016). *" An Improved DCT Based Color Image Watermarking Scheme"*, International Symposium on Mechanical Engineering and Material Science, Advances in Engineering Research, volume 93.