Multimed Tools Appl (2015) 74:10883–10921
DOI 10.1007/s11042-014-2212-y

Image watermarking in real oriented wavelet transform domain

Himanshu Agarwal · Pradeep K. Atrey · Balasubramanian Raman

Received: 11 December 2013 / Revised: 19 July 2014 / Accepted: 22 July 2014 /
Published online: 15 August 2014

Abstract In this paper, we propose blind and non-blind watermarking schemes in the real oriented wavelet transform (ROWT) domain. The ROWT, which is a member of the dual tree complex wavelet transform (DTCWT) family, is chosen as a watermarking domain since the DTCWT has recently emerged as an important new image processing tool. Existing watermarking schemes based on the DTCWT usually lack high embedding capacity. This is mainly due to the fact that the left inverse and the right inverse of the DTCWT (including the ROWT) are not equal. We have observed a relation when the ROWT follows its left inverse, and have used this relation to develop two watermarking schemes in the ROWT domain. Experimental results show that the proposed ROWT based watermarking schemes not only have a much higher capacity than the existing DTCWT based watermarking schemes, but are also robust to various image modification operations such as cropping, Gaussian filter, Gaussian noise, and salt and pepper noise.

Keywords Blind watermarking scheme · Non-blind watermarking scheme · Quotient remainder theorem · Real oriented wavelet transform · Dual tree complex wavelet transform

H. Agarwal (✉)
Department of Mathematics, Indian Institute of Technology Roorkee, Roorkee 247667, India
e-mail: him11dma@iitr.ac.in

P. K. Atrey
Department of Computer Science, College of Computing and Information,
State University of New York, Albany, NY, USA
e-mail: pkatrey@gmail.com

B. Raman
Department of Computer Science and Engineering, Indian Institute of Technology Roorkee,
Roorkee 247667, India
e-mail: balarfma@iitr.ac.in

1 Introduction

Digital watermarking is an effective tool in multimedia that provides solution for digital rights such as broadcast monitoring, piracy, owner identification, copy right protection, proof of ownership, media authentication, fingerprinting/transaction tracking, copy control, legacy enhancement etc. [1, 2, 5, 7, 12, 16, 21, 35, 41]. Recently, digital watermarking has been also used to enhance security of biometric systems [13, 23].

The main idea in digital watermarking is that a watermark (which may be a binary image, a gray scale image, a signature etc.) is embedded into a multimedia (which may be an image, a video, an audio etc.) to obtain a watermarked media. The watermarked media is made available in public domain. To solve digital right claims associated with a watermarked media available in public domain, a watermark is extracted/detected from the watermarked media and the extracted watermark is compared with all possible embedded watermarks using a comparator function to identify an embedded watermark. A comparator function consists of two main components- a similarity measure function and a threshold value. Similarity measure function returns similarity value between two watermarks. If a similarity value is greater than the threshold then watermarks are matched, otherwise not. Maximization of accurate identification rate (which directly depends on the capacity/size/length of watermark) is a common goal in all watermarking applications. If very intelligent adversary with an aim to defeat a watermarking system is present in public domain, then watermarking problems are very challenging.

A watermarked media available in public domain may be attacked by adversary or innocent/common multimedia processing operations. Robust watermarking are designed such that after any kind of attack on the watermarked media, the watermark is successfully identified. Such watermarking schemes are applicable for copyright protection, owner identification etc. Fragile watermarking are designed such that if watermarked media is attacked then watermark is not identified. Such watermarking schemes are helpful in tamper detection. Semi-fragile watermarking schemes are robust against adversary attack and fragile against innocent/common multimedia processing operations [5].

Depending on the required information of original data (original media/cover work and original watermark) in watermark extractor, watermarking schemes are divided into three categories, namely blind (oblivious or public), non-blind (non-oblivious or private) and semi-blind watermarking schemes [1]. Watermarking schemes that do not require the information of original data in their watermark extractor are called blind watermarking schemes. Non-blind watermarking schemes require complete information of original data in their watermark extractor while, semi-blind watermarking schemes need a part of information of original data in their watermark extractor. Usually, non-blind watermarking scheme are more robust than blind watermarking schemes. Non-blind watermarking schemes are applicable in those scenarios, where automated search of original media is possible [19]. However, blind watermarking schemes are more attractive, since, in many watermarking applications, original media can not be made available at watermark extractor [1, 10, 20, 27].

According to domain in which watermark is embedded, watermarking schemes are classified into two broad categories: spatial-domain and transformed-domain schemes. In spatial domain watermarking schemes [13], [11] the watermark is embedded by directly modifying the pixel values of an original media. The main idea in transformed-domain watermarking schemes is that an original media is transformed, transformed coefficients of the original media are modified and the inverse transform is applied on the updated transformed coefficients to obtain the watermarked media. Some popular transformed domain for digital watermarking are discrete cosine transform (DCT) [28], discrete wavelet transform (DWT) [19], discrete Fourier transform (DFT) [38], singular value decomposition (SVD) [2], Fractional transform [3] etc. A proper domain should be selected for watermarking according to application scenarios, for instance, DCT based watermarking schemes have better performance for JPEG [40] standard images and DWT domain based watermarking schemes have better performance for JPEG 2000 [36] standard images [37]. Therefore, scopes exist to develop improved watermarking schemes in various transform domain.

Recently, transforms of complex wavelet family have emerged as very important multimedia processing tools. Complex wavelet integrates the phase concept of Fourier transform and multi-resolution analysis concept of wavelets. DTCWT (dual-tree-complex-wavelet-transform) is an important member of complex wavelet transforms family. An extended version of DTCWT for images is called ROWT (real-oriented-wavelet-transform) [33]. The details of complex wavelets and ROWT are discussed in Section 2. Applications of complex wavelet transforms family have been found in estimating image geometrical structures, local displacement and motion estimation [24, 31, 33], denoising [44], image segmentation [34], seismic imaging [26], disparity estimation [17] and content based image retrieval [8, 14, 15]. In watermarking field, researchers have developed various watermarking schemes in a domain of complex wavelet transform (CWT) family [4, 22, 39, 43]. However, small watermark length/size is a major limitation in all the existing CWT family based watermarking schemes. A main reason for this limitation may be that the left inverse and the right inverse of family members of CWT are not equal.

In this paper, a relation has been found when the ROWT follows its left inverse. Based on this relation, two watermarking schemes have been developed in the ROWT domain for images. One is the non-blind watermarking scheme and other is the blind watermarking scheme. Exploring the observed relation in the proposed non-blind watermarking scheme was easy. However, doing so in the proposed blind watermarking scheme was proved to be difficult. This challenge has been solved by a deep investigation of the Quotient-Remainder theorem for the real numbers. The proposed watermarking schemes have been compared with the existing CWT family based watermarking schemes and a drastic increase in the length/size of watermarks has been shown in both the proposed watermarking schemes. In the experiments, meaningful binary logo watermarks have been used in the proposed watermarking schemes. The performance of the proposed watermarking schemes have been studied under various common image processing operations such as cropping, Gaussian filtering, Gaussian noise and salt and pepper noise. Experimental results demonstrate that the proposed watermarking schemes are more robust than the existing CWT family based watermarking schemes.

The rest of the paper is organized as follows. In Section 2, the ROWT, its implementation on an image and its observed property are discussed.

The proposed watermarking schemes are described in Section 3. Experiments results are analyzed in Section 4. Conclusions are provided in Section 5.

2 The ROWT and its observed property

In this section, ROWT and its implementation on gray scale image are discussed followed by the observed property of the ROWT. A list of main symbols used in this section is defined in Table 1.

Table 1 List of symbols used in Section 2

ROWT	real oriented wavelet transform.
IROWT	left inverse real oriented wavelet transform.
ϕ_h	real scaling function.
ϕ_g	imaginary scaling function.
ψ_h	real wavelet function.
ψ_g	imaginary wavelet function.
h_0	analysis filter corresponding to scaling function ϕ_h.
h_1	analysis filter corresponding to wavelet function ψ_h.
g_0	analysis filter corresponding to scaling function ϕ_g.
g_1	analysis filter corresponding to wavelet function ψ_h.
$\tilde{h}_0$	synthesis filter corresponding to scaling function ϕ_h.
$\tilde{h}_1$	synthesis filter corresponding to wavelet function ψ_h.
$\tilde{g}_0$	synthesis filter corresponding to scaling function ϕ_g.
$\tilde{g}_1$	synthesis filter corresponding to wavelet function ψ_h.

2.1 The ROWT

Selesnick et al. [33] defined the complex wavelet function and the complex scaling function as follows:

$$\psi_c(t) = \psi_h(t) + \mathrm{j}\psi_g(t) \tag{1}$$

$$\phi_c(t) = \phi_h(t) + \mathrm{j}\phi_g(t) \tag{2}$$

where, $\psi_c(t)$ is an analytic function, i.e. $\psi_g(t) = \mathcal{H}[\psi_h(t)]$, $\mathcal{H}[.]$ is the Hilbert transform operator, ϕ_h is the scaling function corresponding to the wavelet ψ_h, ϕ_g is the scaling function corresponding to the wavelet ψ_g and j is the square root of -1.

The ROWT consists of six real-oriented 2D wavelets that are defined as follows,

$$\psi_l(x, y) = \frac{1}{\sqrt{2}}(\psi_{1,l}(x, y) + \psi_{2,l}(x, y)), \tag{3}$$

$$\psi_{l+3}(x, y) = \frac{1}{\sqrt{2}}(\psi_{1,l}(x, y) - \psi_{2,l}(x, y)), \tag{4}$$

where $l = 1, 2, 3$,

$$\psi_{1,1}(x, y) = \phi_h(x)\psi_h(y), \quad \psi_{2,1}(x, y) = \phi_g(x)\psi_g(y),$$
$$\psi_{1,2}(x, y) = \psi_h(x)\phi_h(y), \quad \psi_{2,2}(x, y) = \psi_g(x)\phi_g(y),$$
$$\psi_{1,3}(x, y) = \psi_h(x)\psi_h(y), \quad \psi_{2,3}(x, y) = \psi_g(x)\psi_g(y),$$

and two scaling functions that are defined as follows,

$$\phi_1(x, y) = \phi_h(x)\phi_h(y), \quad \phi_2(x, y) = \phi_g(x)\phi_g(y).$$

The factor $1/\sqrt{2}$ in (3)–(4) is the normalization factor that is used to make the sum/difference operation an orthonormal operation. The ROWT have been implemented on an image using two fast-wavelet-transforms (FWT) [9, 25, 33] in parallel. The analysis filter bank (Fig. 1) of the ROWT is used to decompose an image into eight sub-bands and synthesis filter bank (Fig. 2) of the ROWT is used to reconstruct back the image from

Algorithm 1 Algorithm for 1-level forward ROWT

```
Input: An image $I$ of size $M \times N$ pixels, analysis filters $h_0$, $h_1$, $g_0$, $g_1$.
(∗ See table 1 for detailed explanation of symbols. ∗)
Output: decomposed sub-bands, $A_1$, $A_2$, $P_H$, $P_V$, $P_D$, $N_H$, $N_V$ and $N_D$ each
    of size $\frac{M}{2} \times \frac{N}{2}$ pixels.
1. Convolute $I$ in parallel with $h_0$, $h_1$, $g_0$ and $g_1$ along columns followed by
   down-sampling of factor 2 along columns. Store the results as $I_1$, $I_2$, $I_3$
   and $I_4$ respectively.
2. Convolute $I_1$ in parallel with $h_0$ and $h_1$ along rows followed by down-
   sampling of factor 2 along rows. Store the results as $I_{Ah}$ and $I_{Hh}$ respectively.
3. Convolute $I_2$ in parallel with $h_0$ and $h_1$ along rows followed by down-
   sampling of factor 2 along rows. Store the results as $I_{Vh}$ and $I_{Dh}$ respectively.
4. Convolute $I_3$ in parallel with $g_0$ and $g_1$ along rows followed by down-
   sampling of factor 2 along rows. Store the results as $I_{Ag}$ and $I_{Hg}$ respectively.
5. Convolute $I_4$ in parallel with $g_0$ and $g_1$ along rows followed by down-
   sampling of factor 2 along rows. Store the results as $I_{Vg}$ and $I_{Dg}$ respectively.
6. Assign:
       $A_1 \leftarrow I_{Ah};$
       $A_2 \leftarrow I_{Ag}.$
7. Compute:
       $P_H = \frac{1}{\sqrt{2}}(I_{Hh} + I_{Hg});$
       $N_H = \frac{1}{\sqrt{2}}(I_{Hh} - I_{Hg});$
       $P_V = \frac{1}{\sqrt{2}}(I_{Vh} + I_{Vg});$
       $N_V = \frac{1}{\sqrt{2}}(I_{Vh} - I_{Vg});$
       $P_D = \frac{1}{\sqrt{2}}(I_{Dh} + I_{Dg});$
       $N_D = \frac{1}{\sqrt{2}}(I_{Dh} - I_{Dg}).$
8. return Eight decomposed sub-bands $A_1$, $A_2$, $P_H$, $P_V$, $P_D$, $N_H$, $N_V$ and
   $N_D$ each of size $\frac{M}{2} \times \frac{N}{2}$ pixels.
```

its decomposed sub-bands. The details of the one-level forward ROWT and the one-level inverse ROWT are discussed in the algorithm 1 and the algorithm 2 respectively.

One-level implementation of the ROWT [32, 33] on a sample gray scale image is shown in Fig. 3. The size of the sample image is 256 × 256 pixels and size of the each decomposed sub-band is 128 × 128 pixels. Note that the sub-bands A_1 and A_2 are called approximation

Algorithm 2 Algorithm for 1-level inverse ROWT

```
Input: Eight decomposed sub-bands $A_1$, $A_2$, $P_H$, $P_V$, $P_D$, $N_H$, $N_V$ and $N_D$
    each of size $\frac{M}{2} \times \frac{N}{2}$ pixels, synthesis filters $\tilde{h}_0$, $\tilde{h}_1$, $\tilde{g}_0$, $\tilde{g}_1$.
(* See table 1 for detailed explanation of symbols. *)
Output: Reconstructed image $I$ of size $M \times N$ pixels.
1.  Compute:
        $I_{Hh} = \frac{1}{\sqrt{2}}(P_H + N_H);$
        $I_{Hg} = \frac{1}{\sqrt{2}}(P_H - N_H);$
        $I_{Vh} = \frac{1}{\sqrt{2}}(P_V + N_V);$
        $I_{Vg} = \frac{1}{\sqrt{2}}(P_H - N_H);$
        $I_{Dh} = \frac{1}{\sqrt{2}}(P_D + N_D);$
        $I_{Dg} = \frac{1}{\sqrt{2}}(P_D - N_D).$
2.  Assign:
        $I_{Ah} \leftarrow A_1$
        $I_{Ag} \leftarrow A_2.$
3.  Convolute $I_{Hh}$ and $I_{Dh}$ with $\tilde{h}_1$ along rows succeeded by up-sampling of
    factor 2 along rows. Store the results as $I_1$ and $I_2$ respectively.
4.  Convolute $I_{Hg}$ and $I_{Dg}$ with $\tilde{g}_1$ along rows succeeded by up-sampling of
    factor 2 along rows. Store the results as $I_3$ and $I_4$ respectively.
5.  Convolute $I_{Vh}$ and $I_2$ with $\tilde{h}_1$ along columns succeeded by up-sampling
    of factor 2 along columns. Store the results as $I_5$ and $I_6$ respectively.
6.  Convolute $I_{Vg}$ and $I_4$ with $\tilde{g}_1$ along columns succeeded by up-sampling
    of factor 2 along columns. Store the results as $I_7$ and $I_8$ respectively.
7.  Convolute $I_{Ah}$ and $I_1$ with $\tilde{h}_0$ along columns succeeded by up-sampling
    of factor 2 along columns. Store the results as $I_9$ and $I_{10}$ respectively.
8.  Convolute $I_{Ag}$ and $I_3$ with $\tilde{g}_0$ along columns succeeded by up-sampling
    of factor 2 along columns. Store the results as $I_{11}$ and $I_{12}$ respectively.
9.  Convolute $I_5$ and $I_9$ with $\tilde{h}_0$ along rows succeeded by up-sampling of
    factor 2 along rows. Store the results as $I_{13}$ and $I_{14}$ respectively.
10. Convolute $I_7$ and $I_{11}$ with $\tilde{g}_0$ along rows succeeded by up-sampling of
    factor 2 along rows. Store the results as $I_{15}$ and $I_{16}$ respectively.
11. Compute $y_1 = I_6 + I_{10} + I_{13} + I_{14}$.
12. Compute $y_2 = I_8 + I_{12} + I_{15} + I_{16}$.
13. Compute $I = (y_1 + y_2)/\sqrt{2}$.
```

sub-bands, the sub-bands P_V, P_H and P_D are called positive sub-bands, and the sub-bands N_V, N_H and N_D are called negative sub-bands.

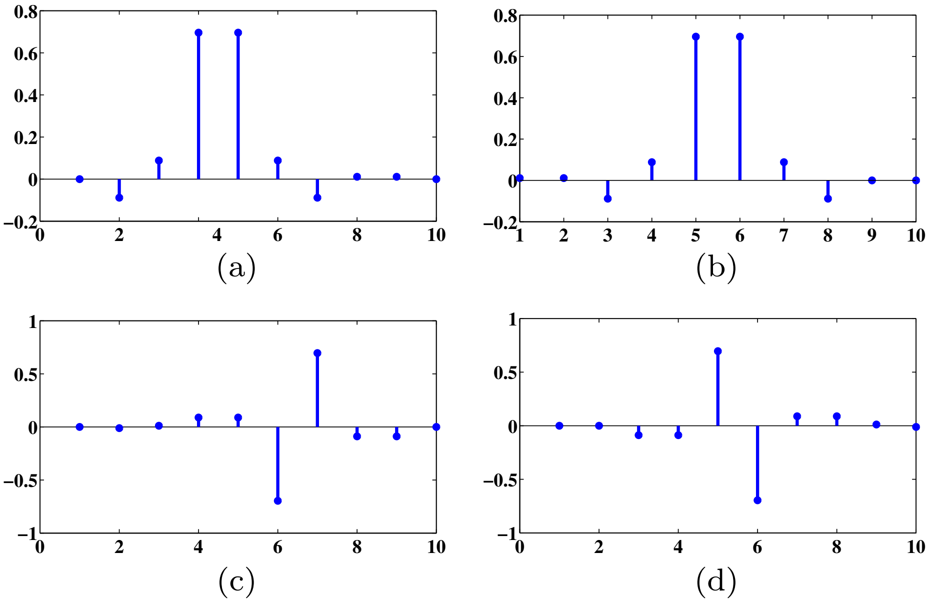


Fig. 1 Analysis filter bank for decomposition of image/signal [32, 33]. **a** Real scaling analysis filter h_0 corresponding to ϕ_h. **b** Imaginary scaling analysis filter g_0 corresponding to ϕ_g. **c** Real wavelet analysis filter h_1 corresponding to ψ_h. **d** Imaginary wavelet analysis filter g_1 corresponding to ψ_g

2.2 Observed property of the ROWT

Let A_1 and A_2 be the approximation sub-bands and $P_\theta s$ and $N_\theta s$ be the positive and negative sub-bands respectively obtained from an image I on applying the ROWT [32, 33] on it , where $\theta = H$, V and D. Modify $P_\theta s$ and $N_\theta s$ as follows:

$$P'_\theta = P_\theta + Q_1, \tag{5}$$

$$N'_\theta = N_\theta + Q_2, \tag{6}$$

where, Q_1 and Q_2 are two real matrices that have a size equal to the size of the P_θs and N_θs. Let I' be the image reconstructed by applying the inverse ROWT on A_1, A_2, P'_θs and N'_θs sub-bands. Let P''_θs and N''_θs be the positive and negative sub-bands respectively obtained from I' by applying the ROWT on it. Then the following two relations have been observed:

$$P''_\theta + N''_\theta = P_\theta + N_\theta + Q_1 \text{if } Q_2 = Q_1, \tag{7}$$

$$P''_\theta - N''_\theta = P_\theta - N_\theta + Q_1 \text{if } Q_2 = -Q_1. \tag{8}$$

A numerical example is provided in Fig. 4 to elaborate and verify the observed property of the ROWT. Figure 4 reports a small mismatch in the observed property. This small mismatch may be due to the truncation error in data representation and transform implementation. Note that inverse ROWT followed by ROWT does not make an identity transform. This situation is very different than the DCT, DWT and other transforms, wherein, inverse of a transform followed by itself makes an identity transform. This main

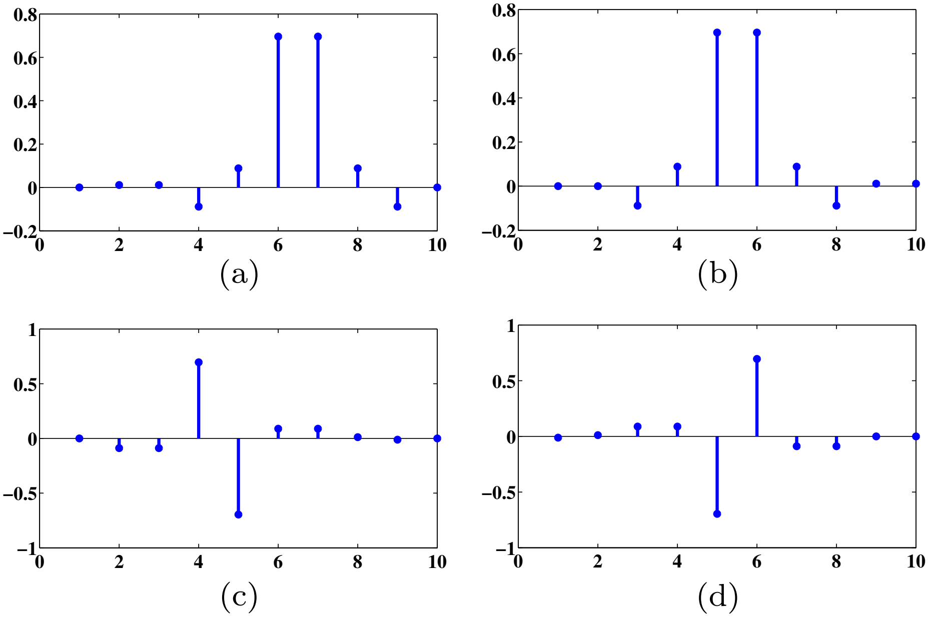


Fig. 2 Synthesis filter bank for reconstruction of image/signal [32, 33]. **a** Real scaling synthesis filter $\tilde{h}_0$ corresponding to ϕ_h. **b** Imaginary scaling synthesis filter $\tilde{g}_0$ corresponding to ϕ_g. **c** Real wavelet synthesis filter $\tilde{h}_1$ corresponding to ψ_h. **d** Imaginary wavelet synthesis filter $\tilde{g}_1$ corresponding to ψ_g

difference between the ROWT and other transforms emphasizes the need for significant changes to the conventional watermarking models. In this paper, the observed property of the ROWT is used as a building block in the proposed watermarking schemes.

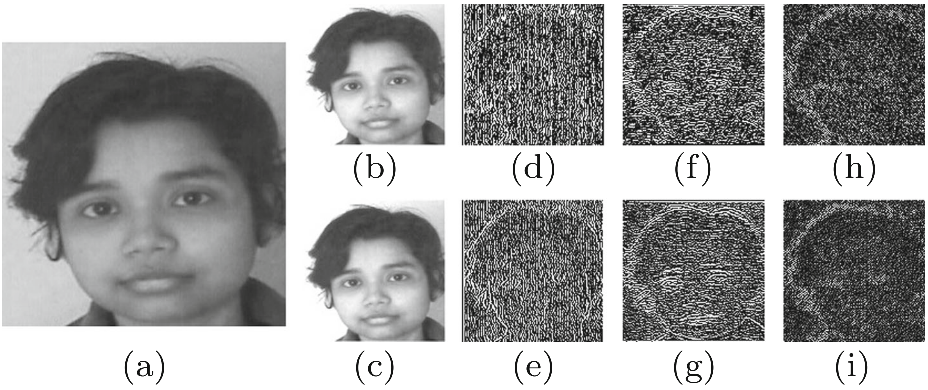
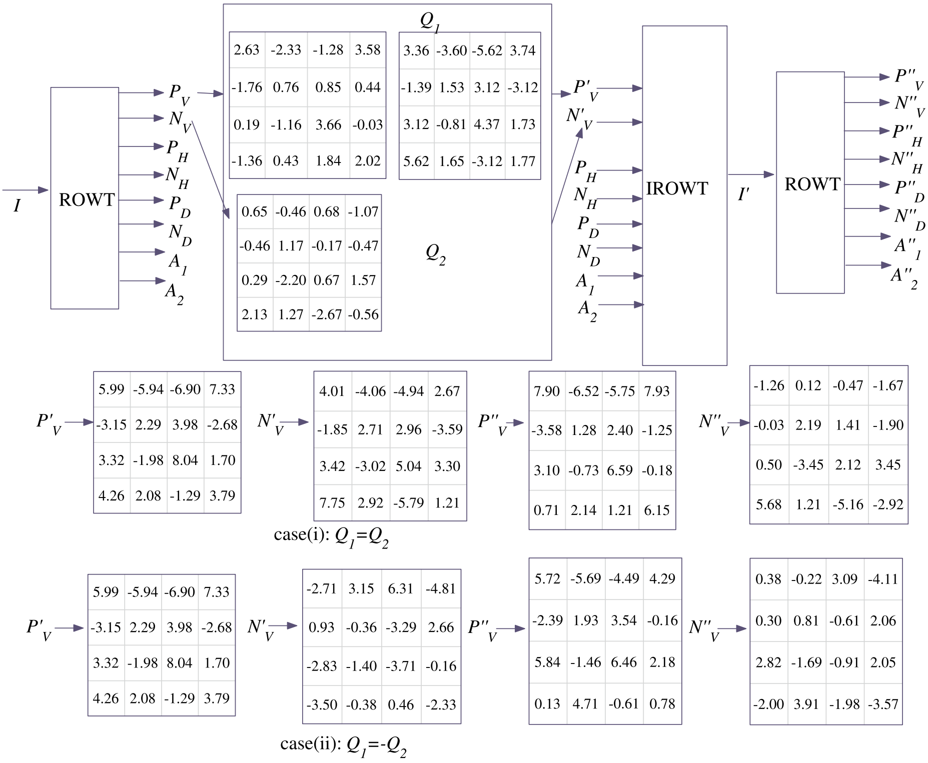


Fig. 3 The ROWT of an image (**a**). **a**: a sample image, **b**: sub-band A_1 corresponding to ϕ_1, **c**: sub-band A_2 corresponding to ϕ_2, **d**: sub-band P_V corresponding to ψ_1, **e**: sub-band N_V corresponding to ψ_4, **f**: sub-band P_H corresponding to ψ_2, **g**: sub-band N_H corresponding to ψ_5, **h**: sub-band P_D corresponding to ψ_3 **i**: sub-band N_D corresponding to ψ_6



P_V

2.63	-2.33	-1.28	3.58
-1.76	0.76	0.85	0.44
0.19	-1.16	3.66	-0.03
-1.36	0.43	1.84	2.02

Q_1

3.36	-3.60	-5.62	3.74
-1.39	1.53	3.12	-3.12
3.12	-0.81	4.37	1.73
5.62	1.65	-3.12	1.77

Q_2

0.65	-0.46	0.68	-1.07
-0.46	1.17	-0.17	-0.47
0.29	-2.20	0.67	1.57
2.13	1.27	-2.67	-0.56

case(i): $Q_1=Q_2$

P'_V

5.99	-5.94	-6.90	7.33
-3.15	2.29	3.98	-2.68
3.32	-1.98	8.04	1.70
4.26	2.08	-1.29	3.79

N'_V

4.01	-4.06	-4.94	2.67
-1.85	2.71	2.96	-3.59
3.42	-3.02	5.04	3.30
7.75	2.92	-5.79	1.21

P''_V

7.90	-6.52	-5.75	7.93
-3.58	1.28	2.40	-1.25
3.10	-0.73	6.59	-0.18
0.71	2.14	1.21	6.15

N''_V

-1.26	0.12	-0.47	-1.67
-0.03	2.19	1.41	-1.90
0.50	-3.45	2.12	3.45
5.68	1.21	-5.16	-2.92

case(ii): $Q_1=-Q_2$

P'_V

5.99	-5.94	-6.90	7.33
-3.15	2.29	3.98	-2.68
3.32	-1.98	8.04	1.70
4.26	2.08	-1.29	3.79

N'_V

-2.71	3.15	6.31	-4.81
0.93	-0.36	-3.29	2.66
-2.83	-1.40	-3.71	-0.16
-3.50	-0.38	0.46	-2.33

P''_V

5.72	-5.69	-4.49	4.29
-2.39	1.93	3.54	-0.16
5.84	-1.46	6.46	2.18
0.13	4.71	-0.61	0.78

N''_V

0.38	-0.22	3.09	-4.11
0.30	0.81	-0.61	2.06
2.82	-1.69	-0.91	2.05
-2.00	3.91	-1.98	-3.57

Fig. 4 Illustration of the observed property of the ROWT using numerical examples

3 Watermarking schemes

This section is divided into two subsections: non-blind watermarking schemes (Section 3.1) and blind watermarking schemes (Section 3.2). In Section 3.1, a traditional non-blind watermarking scheme, proposed non-blind watermarking scheme in the ROWT domain and watermark estimation rules from extracted watermark bits are discussed. In Section 3.2, a traditional blind watermarking scheme and proposed blind watermarking scheme in the ROWT domain are discussed. A list of symbols used in Section 3 is explained in Table 2.

3.1 Non-blind watermarking schemes

A watermark embedding algorithm and corresponding watermark extraction algorithm of a traditional non-blind watermarking scheme are explained in algorithm 7 and algorithm 8 respectively. A numerical example is provided in Fig. 5 to illustrate the traditional non-blind watermarking scheme.

Figure 6a and b summarize the components of the proposed non-blind watermark embedding and watermark extraction algorithms. The details of the watermark embedding and watermark extraction algorithms are discussed in algorithm 9 and algorithm 10. In the algorithm 9, watermark bits are embedded by utilizing (5) and (6), and in the algorithm 10, watermark bits are extracted by utilizing (7) and (8). A numerical example is provided in Fig. 7 to illustrate the proposed non-blind watermarking scheme.

Table 2 List of symbols used in Section 3

I_o	Original image
W_o	Original watermark
α	Watermarking strength
δ	error controller, a parameter in the proposed blind watermark embedding algorithm, which controls error rate in extracted watermarks.
k	$\in \{1, 2\}$, used as a sub-key in the proposed watermarking algorithm.
G	a map such that $G : \{P, N\} \to \{-1, 1\}$, used as a sub-key in the proposed watermarking algorithm.
P, N (regular text/ superscript/ subscript)	represent a positive, a negative sub-band respectively of a ROWT transformed image.
Subscripts H, V, D	represent a horizontal, a vertical, a detailed sub-band respectively of a ROWT transformed image.
A (regular text or superscript)	represents an approximate sub-band of a ROWT transformed image.
I_W	Watermarked image
I_w	an image, from which watermark is to be extracted (may be watermarked, attacked, unmarked).
W	a temporary variable used to store extracted bits.
$W_e/W_e^i s$	Extracted watermark(s)
$M_1 \times N_1$	size of original/watermarked image (pixels)
$M_2 \times N_2$	size of original/extracted watermark (pixels)
L_1	A set of watermarking pixels from a domain (spatial/transformed) of the I_o, where, watermark is embedded/ from I_W/I_w, from where, watermark is to be extracted.
L_2	Set of all pixels in the $W_o/W_e/W_e^i s$.
l_1	a pixel from the L_1.
l_2	a pixel from the L_2 corresponds to the l_1.
π	a map that associates each l_1 with a l_2.
$I_o(l_1)$	the pixel/coefficient value of the I_o at the l_1.
$W_o(l_2)$	the pixel value of the W_o at the l_2.
$I_W(l_1)$	pixel/coefficient value of the I_W at the l_1.
$W_e(l_2)$	pixel value of the W_e at the l_2.

According to the traditional non-blind watermarking scheme (Algorithms 7 and 8), one watermark bit is embedded at one position and one watermark bit is extracted using one position. According to the proposed non-blind watermarking scheme (Algorithms 9 and 10), one watermark bit is embedded at two positions and one watermark bit is extracted using two positions. This is the main difference between the traditional non-blind watermarking scheme and the proposed non-blind watermarking scheme.

A watermark is estimated from the extracted watermark bits. In watermark estimation, the following three scenarios can arise:

Algorithm 3 A traditional non-blind watermark embedding algorithm in transform domain [6, 19]

```
Input: $I_o$, $W_o$, $\alpha$, $L_1$, $\pi$, and a transform $T$ and its inverse $T^{-1}$.
(* See table 2 for detailed explanation of each symbol. *)
Output: Watermarked image $I_W$.
1.  Apply transform $T$ on the $I_o$ to obtain the transformed image $TI_o$.
2.  Define $TI_W = TI_o$, where, $TI_W$ represents an intermediate watermarked
    image $I_W$ in the transform domain $T$.
3.  Select a $l_1 \in L_1$.
4.  Compute $l_2 = \pi(l_1)$.
5.  Compute and assign $TI_W(l_1) \leftarrow TI_o(l_1) + \alpha W(l_2)$.
(* one watermark bit is embedded at one position *)
6.  Repeat steps 4 to 5 for all values of $l_1$.
7.  Apply inverse $T$ transform ($T^{-1}$) on the updated $TI_W$ to obtain the
    watermarked image $I_W$.
8.  return Watermarked image $I_W$.
```

1. A watermark is embedded once and each watermark bit is embedded once. In this case, π is an one-to-one function and one watermark W_e is estimated as follows:

$$W_e(l_2) = W(l_1). \tag{9}$$

Algorithm 4 A traditional non-blind watermark extraction algorithm in transform domain [6, 19]

```
Input: $I_o$, $I_w$, $\alpha$, $L_1$, $\pi$, transform $T$ same as used in the algorithm 3, and a
    watermark estimation rule.
(* See table 2 for detailed explanation of each symbol. *)
Output: $W_e/W_e^i s$.
1.  Apply transform $T$ on the $I_w$ and the $I_o$ to obtain the transformed image
    $TI_w$ and $TI_o$ respectively.
2.  Select a $l_1 \in L_1$.
3.  Compute $d = |TI_w(l_1) - TI_o(l_1)|$.
4.  Extract a bit and store in $W$ as follows:
        $W(l_1) = \begin{cases} 0 \text{ if } d < \alpha/2 \\ 1 \text{ if } d \geq \alpha/2 \end{cases}.$    (9)
(* one watermark bit is extracted using one position *)
5.  Repeat steps 3 to 4 for all values of $l_1$.
6.  Use the watermark estimation rule to estimate extracted watermark(s)
    $W_e/W_e^i s$ from the $W$.
7.  return $W_e/W_e^i s$.
(* Extracted watermark(s) *).
```

Algorithm 5 The proposed non-blind watermark embedding algorithm in the ROWT domain

```
Input: $I_o$, $W_o$, $\alpha$, $L_1$, $\pi$, $G(P)$ and $G(N)$.
(* See table 2 for detailed explanation of each symbol. *)
Output: Watermarked image $I_W$.
1.  Apply the 1-level ROWT on the $I_o$. Store the decomposed sub-bands as
    $I_oA_1$, $I_oA_2$, $I_oP_H$, $I_oP_V$, $I_oP_D$, $I_oN_H$, $I_oN_V$, $I_oN_D$.
2.  Define $I_oP_HW = I_oP_H$, $I_oP_VW = I_oP_V$, $I_oP_DW = I_oP_D$, $I_oN_HW =$
    $I_oN_H$, $I_oN_VW = I_oN_V$, $I_oN_DW = I_oN_D$.
3.  Compute $s = G(P) \times G(N)$.
4.  Select a $l_1 \in L_1$.
5.  Extract $\theta \in \{H, V, D\}$ from $l_1$.
6.  Compute $l_2 = \pi(l_1)$.
7.  Assign:
        $I_oP_\theta W(l_1) \leftarrow I_oP_\theta(l_1) + \alpha \times G(P) \times W_o(l_2);$
        $I_oN_\theta W(l_1) \leftarrow I_oN_\theta(l_1) + \alpha \times G(N) \times W_o(l_2).$
(* one watermark bit is embedded at two positions *)
8.  Repeat steps 5 to 7 for all values of $l_1$.
9.  Apply inverse ROWT on the sub-bands $I_oA_1$, $I_oA_2$, and updated $I_oP_HW$,
    $I_oP_VW$, $I_oP_DW$, $I_oN_HW$, $I_oN_VW$, $I_oN_DW$ to obtain the watermarked
    image $I_W$.
10. return $I_W$.
(* Watermarked image *)
```

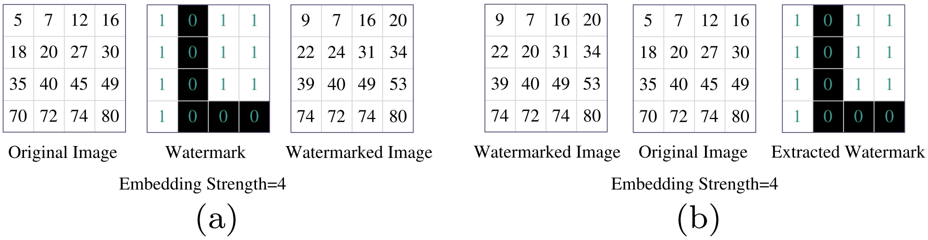
2. A watermark is embedded n (> 1) times and each watermark bit is embedded n times. In this scenario, π is a many-to-one function, the redundancy of the watermark is n, and n watermarks W_e^i are estimated as follows:

$$W_e^i(l_2) = W(\pi^{-1}(\pi(l_1))_i), \tag{10}$$

where, $i = 1, 2, \cdots n$, π^{-1} is the inverse map of π and

$$\pi^{-1}(\pi(l_1)) = \{\pi^{-1}(\pi(l_1))_i\}.$$

3. A watermark is embedded once and each watermark bit is embedded n (an odd number) times. In this situation, π is a many-to-one function, the redundancyof



Original Image

5	7	12	16
18	20	27	30
35	40	45	49
70	72	74	80

Watermark

1	0	1	1
1	0	1	1
1	0	1	1
1	0	0	0

Watermarked Image

9	7	16	20
22	24	31	34
39	40	49	53
74	72	74	80

Embedding Strength=4

(a)

Watermarked Image

9	7	16	20
22	20	31	34
39	40	49	53
74	72	74	80

Original Image

5	7	12	16
18	20	27	30
35	40	45	49
70	72	74	80

Extracted Watermark

1	0	1	1
1	0	1	1
1	0	1	1
1	0	0	0

Embedding Strength=4

(b)

Fig. 5 A numerical example to illustrate a traditional non-blind watermarking algorithms. **a** Embedding algorithm. **b**: Extraction algorithm

Algorithm 6 The proposed non-blind watermark extraction algorithm in the ROWT domain

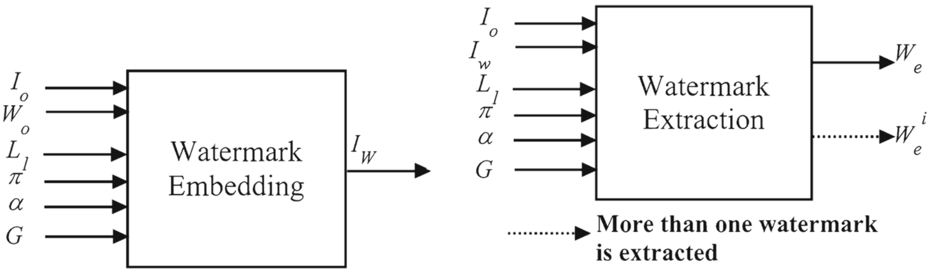
```
Input: $I_w$, a watermark estimation rule, and $I_o$, $\alpha$, $L_1$, $\pi$, $G(P)$, $G(N)$ same as used in algorithm 5.
(* See table 2 for detailed explanation of each symbol. *)
Output: $W_e/W_e^i s$
1. Apply the 1-level ROWT on the $I_o$ and $I_W$. Store the decomposed sub-bands of $I_o$ as $I_oA_1$, $I_oA_2$, $I_oP_H$, $I_oP_V$, $I_oP_D$, $I_oN_H$, $I_oN_V$ and $I_oN_D$, and sub-bands of $I_w$ as $I_wA_1$, $I_wA_2$, $I_wP_H$, $I_wP_V$, $I_wP_D$, $I_wN_H$, $I_wN_V$ and $I_wN_D$.
2. Compute $s = G(P) \times G(N)$.
3. Select a $l_1 \in L_1$.
4. Extract $\theta \in \{H, V, D\}$ from $l_1$.
5. Compute
       $d = \left| \begin{array}{c} [I_wP_\theta(l_1) + s \times I_wN_\theta(l_1)] - \\ [I_oP_\theta(l_1) + s \times I_oN_\theta(l_1)] \end{array} \right|.$   (10)
6. Extract a bit and store it in $W$ as follows:
       $W(l_1) = \begin{cases} 0 \text{ if } d < \alpha/2 \\ 1 \text{ if } d \geq \alpha/2 \end{cases}.$   (11)
(* one watermark bit is extracted using two positions *)
7. Repeat steps 4 to 6 for all values of $l_1$.
8. Use the watermark estimation rule to estimate extracted watermark(s) $W_e/W_e^i s$ from the $W$.
9. return $W_e/W_e^i s$.
(* Extracted watermark(s) *).
```

the watermark bits is *n* and one watermark is estimated by the voting method as follows:

$$W_e(l_2) = b(\pi^{-1}(\pi(l_1))), \tag{11}$$

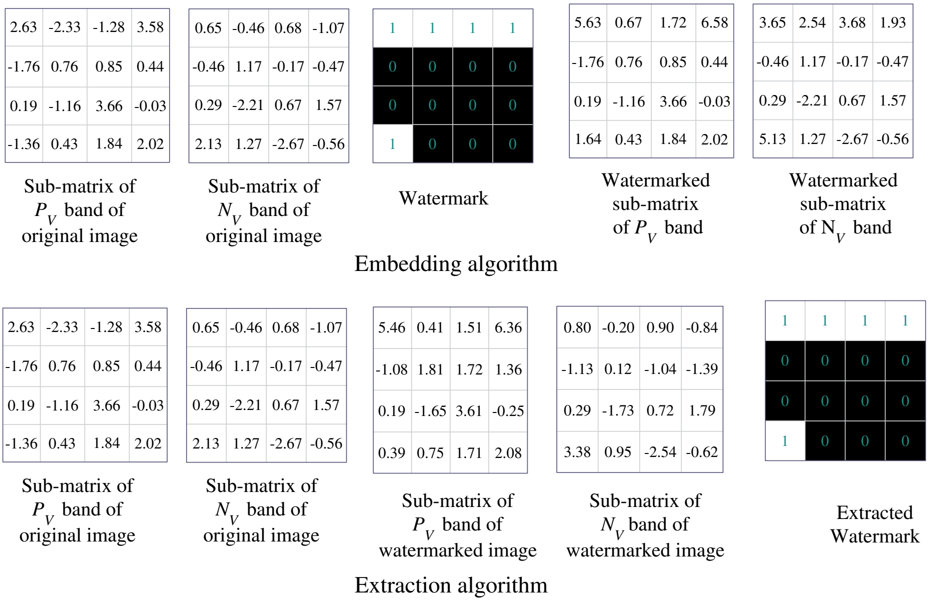
where,

$$b(.) = \begin{cases} 0 \text{ if } c_0(.) > c_1(.) \\ 1 \text{ if } c_0(.) < c_1(.) \end{cases}, \tag{12}$$



(a) Watermark embedding (b) Watermark extraction

Fig. 6 Overview of proposed non-blind watermarking algorithm



Sub-matrix of P_V band of original image

2.63	-2.33	-1.28	3.58
-1.76	0.76	0.85	0.44
0.19	-1.16	3.66	-0.03
-1.36	0.43	1.84	2.02

Sub-matrix of N_V band of original image

0.65	-0.46	0.68	-1.07
-0.46	1.17	-0.17	-0.47
0.29	-2.21	0.67	1.57
2.13	1.27	-2.67	-0.56

Watermark

1	1	1	1
0	0	0	0
0	0	0	0
1	0	0	0

Watermarked sub-matrix of P_V band

5.63	0.67	1.72	6.58
-1.76	0.76	0.85	0.44
0.19	-1.16	3.66	-0.03
1.64	0.43	1.84	2.02

Watermarked sub-matrix of N_V band

3.65	2.54	3.68	1.93
-0.46	1.17	-0.17	-0.47
0.29	-2.21	0.67	1.57
5.13	1.27	-2.67	-0.56

Embedding algorithm

Sub-matrix of P_V band of original image

2.63	-2.33	-1.28	3.58
-1.76	0.76	0.85	0.44
0.19	-1.16	3.66	-0.03
-1.36	0.43	1.84	2.02

Sub-matrix of N_V band of original image

0.65	-0.46	0.68	-1.07
-0.46	1.17	-0.17	-0.47
0.29	-2.21	0.67	1.57
2.13	1.27	-2.67	-0.56

Sub-matrix of P_V band of watermarked image

5.46	0.41	1.51	6.36
-1.08	1.81	1.72	1.36
0.19	-1.65	3.61	-0.25
0.39	0.75	1.71	2.08

Sub-matrix of N_V band of watermarked image

0.80	-0.20	0.90	-0.84
-1.13	0.12	-1.04	-1.39
0.29	-1.73	0.72	1.79
3.38	0.95	-2.54	-0.62

Extracted Watermark

1	1	1	1
0	0	0	0
0	0	0	0
1	0	0	0

Extraction algorithm

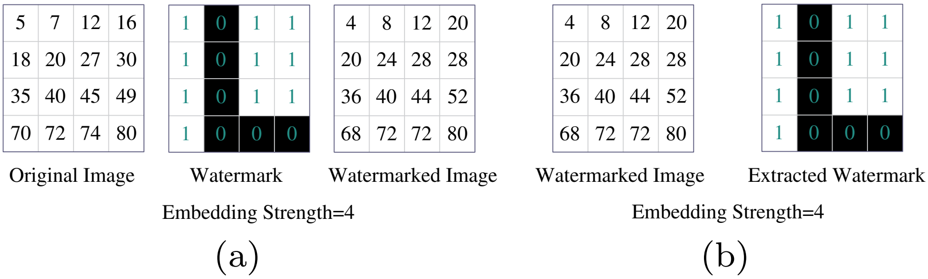
Fig. 7 A numerical example to illustrate the proposed non-blind watermarking scheme in the ROWT domain. In this example, $\alpha = 3$, and $G(P) = G(N) = 1$

and $c_0(\pi^{-1}(\pi(l_1)))$ and $c_1(\pi^{-1}(\pi(l_1)))$ are a number of 0 watermark bits and 1 watermark bits, respectively, at the $\pi^{-1}(\pi(l_1))$.

3.2 Blind watermarking schemes

3.2.1 A traditional blind watermarking scheme

A watermark embedding algorithm and corresponding watermark extraction algorithm of a traditional blind watermarking scheme are explained in Algorithms 3 and 4 respectively. The core idea in the watermark embedding algorithm is that *if watermark bit is one, then make the watermarking coefficient odd multiple of watermarking strength else if watermark bit is zero, then make the watermarking coefficient even multiple of watermarking strength.*



Original Image

5	7	12	16
18	20	27	30
35	40	45	49
70	72	74	80

Watermark

1	0	1	1
1	0	1	1
1	0	1	1
1	0	0	0

Watermarked Image

4	8	12	20
20	24	28	28
36	40	44	52
68	72	72	80

Embedding Strength=4

(a)

Watermarked Image

4	8	12	20
20	24	28	28
36	40	44	52
68	72	72	80

Extracted Watermark

1	0	1	1
1	0	1	1
1	0	1	1
1	0	0	0

Embedding Strength=4

(b)

Fig. 8 A numerical example to illustrate a traditional blind watermarking algorithms. **a** Embedding algorithm. **b**: Extraction algorithm

Algorithm 7 A traditional blind watermark embedding algorithm in transform domain [1, 18, 42]

```
Input: $I_o$, $W_o$, $\alpha$, $L_1$, $\pi$, an arbitrary transform $T$ and its inverse transform
    $T^{-1}$.
(* See table 2 for detailed explanation of each symbol. *)
Output: Watermarked image $I_W$.
1.  Apply transform $T$ on the $I_o$ to obtain the transformed image $TI_o$.
2.  Define $TI_W = TI_o$, where, $TI_W$ represents an intermediate watermarked
    image $I_W$ in the transform domain $T$.
3.  Compute $r = TI_o(l_1) \mod \alpha$.
4.  Compute $b = (TI_o(l_1) - r)/\alpha$.
5.  Compute $l_2 = \pi(l_1)$.
6.  if($b \mod 2 == 0$ AND $W_o(l_2) == 0$) then
        $TI_W(l_1) \leftarrow TI_o(l_1) - r$
7.  else if($b \mod 2 == 1$ AND $W_o(l_2) == 0$) then
        $TI_W(l_1) \leftarrow TI_o(l_1) - r + \alpha$
8.  else if($b \mod 2 == 0$ AND $W_o(l_2) == 1$) then
        $TI_W(l_1) \leftarrow TI_o(l_1) - r + \alpha$
9.  else if($b \mod 2 == 1$ AND $W_o(l_2) == 1$) then
        $TI_W(l_1) \leftarrow TI_o(l_1) - r$
(* one watermark bit is embedded at one position *)
10. end if
11. Repeat steps 3 to 10 for all values of $l_1$.
12. Apply inverse $T$ transform ($T^{-1}$) on the updated $TI_W$ to obtain the
    watermarked image $I_W$.
13. return $I_W$.
(* Watermarked image *)
```

The core idea in the watermark extraction algorithm is that *if watermarked coefficient is odd multiple of watermarking strength, then extracted watermark bit is one else if watermarked coefficient is even multiple of watermarking strength, then extracted watermark bit is zero*. A numerical example is provided in Fig. 8 to illustrate the traditional blind watermarking scheme.

3.2.2 Special mathematical properties

The core theory in the traditional blind watermarking scheme is to make the watermarked coefficients an even/odd multiple of watermarking strength. Success of this theory depends on the fact that watermarked coefficients are well preserved after applying a transform succeeded by its inverse transform. This fact is not true for the ROWT. The core theory in the proposed blind watermarking scheme in the ROWT domain is to add/subtract

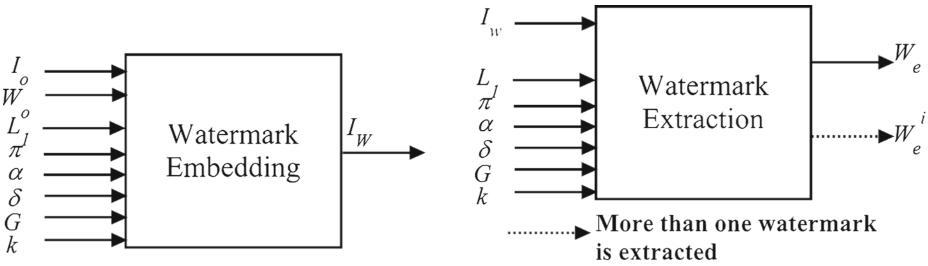
Algorithm 8 A traditional blind watermark extraction algorithm in transform domain [1, 18, 42]

```
Input: $I_w$, watermark estimation rule, and $\alpha$, $L_1$, $\pi$ and transform $T$ same as
    used in the algorithm 7.
(* See table 2 for detailed explanation of each symbol. *)
Output: $W_e/W_e^i s$.
1.  Apply transform $T$ on the $I_w$ to obtain the transformed image $TI_w$.
2.  Compute $r = TI_w(l_1) \mod \alpha$.
3.  if($r > \frac{\alpha}{2}$) then
        $TI_w(l_1) \leftarrow TI_w(l_1) + \alpha - r$
4.  else
        $TI_w(l_1) \leftarrow TI_w(l_1) - r$
5.  end if
6.  Compute $q = TI_w(l_1)/\alpha$.
7.  Compute and store extracted bit in $W$ as:
8.  if$q$ is even then
        $W(l_1) = 0$
9.  else
        $W(l_1) = 1.$
10. end if
(* one watermark bit is extracted using one position *)
11. Repeat steps 2 to 10 for all values of $l_1$.
12. Use the watermark estimation rule to estimate extracted watermark(s)
    $W_e/W_e^i s$ from the $W$.
13. return $W_e/W_e^i s$.
(* Extracted watermark(s) *).
```

a value in coefficients of positive and negative sub-bands of the original image according to the observed property of the ROWT such that remainder of sum of the added value and sum/difference of the coefficients of the positive and negative sub-bands belongs to a well defined desired interval. Success of this theory is guaranteed by the relations (7) and (8). The added values are found using specially derived mathematical properties based on the Quotient-Remainder theorem. The added/subtracted values depend on watermarking strength and a desired interval.

Let μ be a non-negative real number and α be a positive real number. Then the remainder r can be found such that $\mu = \alpha m + r$, where m is the non-negative integer and $0 \leq r < \alpha$. This is the Quotient-Remainder theorem. Here, m is termed as the quotient. Based on the Quotient-Remainder theorem, the following properties hold:

1. remainder(r_1) of $\mu + \frac{\alpha - r}{2}$ is $\frac{\alpha}{2} + \frac{\delta\alpha}{2} < r_1 < \alpha - \frac{\delta\alpha}{2}$, if $\delta\alpha < r < \alpha - \delta\alpha$,
2. remainder(r_2) of $\mu + \frac{\alpha + \delta\alpha}{2}$ is $\frac{\alpha}{2} + \frac{\delta\alpha}{2} \leq r_2 \leq \frac{\alpha}{2} + \frac{3\delta\alpha}{2}$, if $r \leq \delta\alpha$,



(a) Watermark embedding algorithm (b) Watermark extraction algorithm

Fig. 9 Overview of proposed blind watermarking algorithm

3. remainder(r_3) of $\mu + \frac{2\alpha-\delta\alpha}{2}$ is $\frac{\alpha}{2} - \frac{3\delta\alpha}{2} \le r_3 < \alpha - \frac{\delta\alpha}{2}$, if $r \ge \alpha - \delta\alpha$,
4. remainder(r_4) of $\mu + \frac{2\alpha-r}{2}$ is $\frac{\delta\alpha}{2} < r_4 < \frac{\alpha}{2} - \frac{\delta\alpha}{2}$, if $\delta\alpha < r < \alpha - \delta\alpha$,
5. remainder(r_5) of $\mu + \frac{2\alpha+\delta\alpha}{2}$ is $\frac{\delta\alpha}{2} \le r_5 \le \frac{3\delta\alpha}{2}$, if $r \le \delta\alpha$,
6. remainder(r_6) of $\mu + \frac{3\alpha-\delta\alpha}{2}$ is $\frac{\alpha}{2} - \frac{3\delta\alpha}{2} \le r_6 < \frac{\alpha}{2} - \frac{\delta\alpha}{2}$, if $r \ge \alpha - \delta\alpha$,

as long as $\delta \in \left(0, \frac{1}{3}\right)$. Moreover,

7. $\frac{\alpha}{2} + \frac{\delta\alpha}{2} \le r_1, r_2, r_3 < \alpha - \frac{\delta\alpha}{2}$,
8. $\frac{\delta\alpha}{2} \le r_4, r_5, r_6 < \frac{\alpha}{2} - \frac{\delta\alpha}{2}$.

In properties 1–6, the term μ is equivalent to the sum (or difference) of coefficients of the positive and negative sub-bands of an original image in the ROWT domain. Properties 1–6 are used in the proposed blind watermark embedding Algorithm 5 and properties 7–8 are used in the proposed blind watermark extraction Algorithm 6.

3.2.3 The proposed blind watermarking scheme in the ROWT domain

Figure 9a summarizes the components in the watermark embedding algorithm and Fig. 9b summarizes the components in the watermark extraction algorithm. The details of the watermark embedding algorithm and watermark extraction algorithms are discussed in Algorithms 5 and 6 respectively. A numerical example is provided in Fig. 10 to illustrate the proposed blind watermarking scheme.

Like the proposed non-blind watermarking scheme, in this scheme also (Algorithms 5 and 6), one watermark bit is embedded at two positions and one watermark bit is extracted using two positions.

4 Experiments, results, and analysis

We have performed six experiments for a detailed analysis of the proposed watermarking schemes. Experiment 1 tests the proposed watermarking schemes and studies the effect of embedding strength and error controller. This experiment helps to find optimal embedding strength and error controller. Experiment 2 elaborates the importance of the observed property of ROWT. Experiment 3 studies the performance of the proposed watermarking schemes and studies the effect of embedding strength and error controller

Algorithm 9 The proposed blind watermark embedding algorithm in the ROWT domain

```
Input: $I_o$, $W_o$, $\alpha$, $L_1$, $\pi$, $G(P)$, $G(N)$, $\delta$ and $k$.
(* See table 2 for detailed explanation of each symbol. *)
Output: Watermarked image $I_W$.
1.  Apply the 1-level ROWT on the $I_o$. Store the decomposed sub-bands as
    $I_oA_1$, $I_oA_2$, $I_oP_H$, $I_oP_V$, $I_oP_D$, $I_oN_H$, $I_oN_V$, $I_oN_D$.
2.  Define $I_oP_HW = I_oP_H$, $I_oP_VW = I_oP_V$, $I_oP_DW = I_oP_D$, $I_oN_HW =$
    $I_oN_H$, $I_oN_VW = I_oN_V$, $I_oN_DW = I_oN_D$.
3.  Compute $s = G(P) \times G(N)$.
4.  Select a $l_1 \in L_1$.
5.  Compute $l_2 = \pi(l_1)$.
6.  Extract $\theta \in \{H, V, D\}$ from the $l_1$.
7.  Compute $r = |I_oP_\theta(l_1) + s \times I_oN_\theta(l_1)| \mod \alpha$.
8.  Compute $\alpha_1$ as follows:
(* this computation is according to properties 1-6 discussed in section 3.2.2. *)

    $$\alpha_1 = \begin{cases} \alpha + \delta\alpha & \text{if } 0 \le r \le \delta\alpha \\ \alpha - r & \text{if } \delta\alpha < r < \alpha - \delta\alpha \\ 2\alpha - \delta\alpha & \text{if } \alpha - \delta\alpha \le r < \alpha \end{cases}. \qquad (16)$$

9.  Compute $\alpha_2$ as follows:
(* this computation is according to properties 1-6 discussed in section 3.2.2. *)

    $$\alpha_2 = \begin{cases} 2\alpha + \delta\alpha & \text{if } 0 \le r \le \delta\alpha \\ 2\alpha - r & \text{if } \delta\alpha < r < \alpha - \delta\alpha \\ 3\alpha - \delta\alpha & \text{if } \alpha - \delta\alpha \le r < \alpha \end{cases}. \qquad (17)$$

10. if($k == 1$ AND $W_o(l_2) == 0$) then
        $I_oP_\theta W(l_1) = I_oP_\theta(l_1) + G(P) \times \frac{\alpha_1}{2}$;
        $I_oN_\theta W(l_1) = I_oN_\theta(l_1) + G(N) \times \frac{\alpha_1}{2}$;
11. else if($k == 1$ AND $W_o(l_2) == 1$) then
        $I_oP_\theta W(l_1) = I_oP_\theta(l_1) + G(P) \times \frac{\alpha_2}{2}$;
        $I_oN_\theta W(l_1) = I_oN_\theta(l_1) + G(N) \times \frac{\alpha_2}{2}$;
12. else if($k == 2$ AND $W_o(l_2) == 0$) then
        $I_oP_\theta W(l_1) = I_oP_\theta(l_1) + G(P) \times \frac{\alpha_2}{2}$;
        $I_oN_\theta W(l_1) = I_oN_\theta(l_1) + G(N) \times \frac{\alpha_2}{2}$;
13. else if($k == 2$ AND $W_o(l_2) == 1$) then
        $I_oP_\theta W(l_1) = I_oP_\theta(l_1) + G(P) \times \frac{\alpha_1}{2}$;
        $I_oN_\theta W(l_1) = I_oN_\theta(l_1) + G(N) \times \frac{\alpha_1}{2}$;
14. end if
(* one watermark bit is embedded at two positions *)
15. Repeat steps 4 to 14 for all values of $l_1$.
16. Apply inverse ROWT on the sub-bands $I_oA_1$, $I_oA_2$, and updated $I_oP_HW$,
    $I_oP_VW$, $I_oP_DW$, $I_oN_HW$, $I_oN_VW$ and $I_oN_DW$ to obtain the water-
    marked image $I_W$.
17. return $I_W$.
(* Watermarked image *).
```

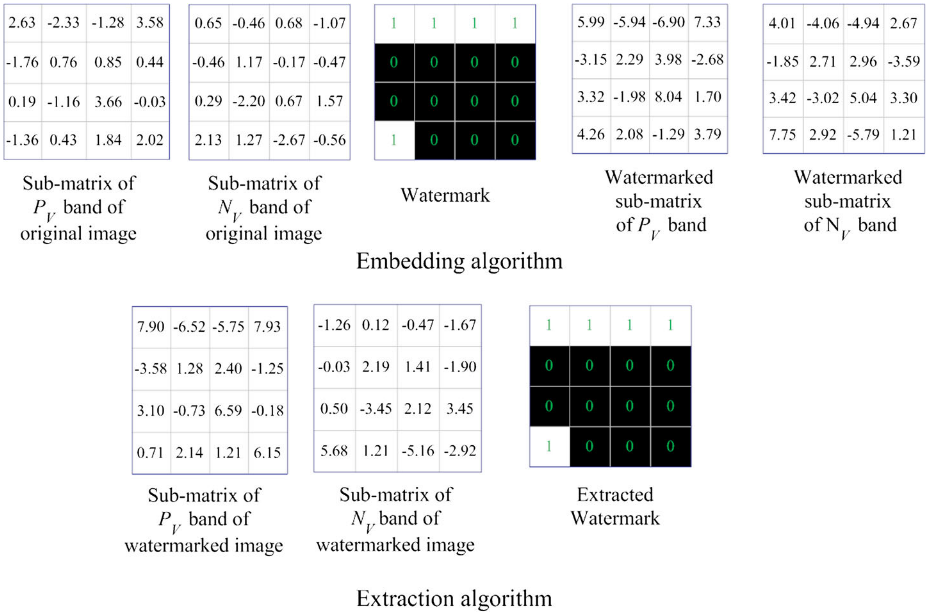
Algorithm 10 The proposed blind watermark extraction algorithm in the ROWT domain

```
Input: $I_w$, watermark estimation rule, and $\alpha$, $L_1$, $\pi$, $G(P)$, $G(N)$, $\delta$ and $k$
    same as used in the algorithm 9.
(* See table 2 for detailed explanation of each symbol. *)
Output: $W_e/W_e^i s$.
1.  Apply ROWT on the $I_w$. Store the decomposed sub-bands as $I_w A_1$, $I_w A_2$,
    $I_w P_H$, $I_w P_V$, $I_w P_D$, $I_w N_H$, $I_w N_V$, $I_w N_D$.
2.  Compute $s = G(P) \times G(N)$.
3.  Extract $\theta \in \{H, V, D\}$ from $l_1$.
4.  Compute $r = |I_o P_\theta(l_1) + s \times I_o N_\theta(l_1)| \mod \alpha$.
5.  Extract a bit and store in $W$ as follows:
6.  if($k == 1$) then
        $W(l_1) = \begin{cases} 1 \text{ if } 0 \leq r < \frac{\alpha}{2} \\ 0 \text{ if } \frac{\alpha}{2} \leq r < \alpha \end{cases};$   (18)
7.  else if($k == 2$) then
        $W(l_1) = \begin{cases} 0 \text{ if } 0 \leq r < \frac{\alpha}{2} \\ 1 \text{ if } \frac{\alpha}{2} \leq r < \alpha \end{cases}.$   (19)
8.  end if
(* one watermark bit is extracted using two positions *)
9.  Repeat Steps 3 to 8 for all values of $l_1$.
10. Use the watermark estimation rule to estimate extracted watermark(s)
    $W_e/W_e^i s$ from the $W$.
11. return $W_e/W_e^i s$.
(* Extracted watermark(s) *).
```

under the formatting operation on the watermarked images. Experiment 4 evaluates the performance of the proposed watermarking schemes under various attacks on formatted watermarked images. Experiment 5 compares the proposed watermarking schemes and existing watermarking schemes without any post operations/attacks on the watermarked images. Experiment 6 compares the performance of the proposed watermarking schemes and existing watermarking schemes under various post operations/attacks on the watermarked images.

In the experiments, we have used a data-set that consists of six host images and five watermarks. Figure 11a shows all host images (h_1, h_2, h_3, h_4, h_5, h_6) and Fig. 11b shows all watermarks (W_1, W_2, W_3, W_4 and W_5) of the data-set. Each host image is an eight bit gray scale image of size 256 × 256 pixels and each watermark is a black and white (binary) image of size 128 × 128 pixels. We have used all the combinations of the host images and the watermarks to obtain different watermarked images.

A common setup in all the experiments is as follows. We have repeated a watermark three times by embedding it once in V, H and D sub-bands of a host image. This makes watermarks of an effective size of $3 \times 128 \times 128$ pixels. L_1 consists of all the locations of V, H, D sub-bands of a host image. π is a simple left-right-top-bottom scan map. We have set $G(P) = G(N) = 1$. For the proposed blind watermarking scheme, we have used $k = 1$.



Sub-matrix of P_V band of original image

2.63	-2.33	-1.28	3.58
-1.76	0.76	0.85	0.44
0.19	-1.16	3.66	-0.03
-1.36	0.43	1.84	2.02

Sub-matrix of N_V band of original image

0.65	-0.46	0.68	-1.07
-0.46	1.17	-0.17	-0.47
0.29	-2.20	0.67	1.57
2.13	1.27	-2.67	-0.56

Watermark

1	1	1	1
0	0	0	0
0	0	0	0
1	0	0	0

Watermarked sub-matrix of P_V band

5.99	-5.94	-6.90	7.33
-3.15	2.29	3.98	-2.68
3.32	-1.98	8.04	1.70
4.26	2.08	-1.29	3.79

Watermarked sub-matrix of N_V band

4.01	-4.06	-4.94	2.67
-1.85	2.71	2.96	-3.59
3.42	-3.02	5.04	3.30
7.75	2.92	-5.79	1.21

Embedding algorithm

Sub-matrix of P_V band of watermarked image

7.90	-6.52	-5.75	7.93
-3.58	1.28	2.40	-1.25
3.10	-0.73	6.59	-0.18
0.71	2.14	1.21	6.15

Sub-matrix of N_V band of watermarked image

-1.26	0.12	-0.47	-1.67
-0.03	2.19	1.41	-1.90
0.50	-3.45	2.12	3.45
5.68	1.21	-5.16	-2.92

Extracted Watermark

1	1	1	1
0	0	0	0
0	0	0	0
1	0	0	0

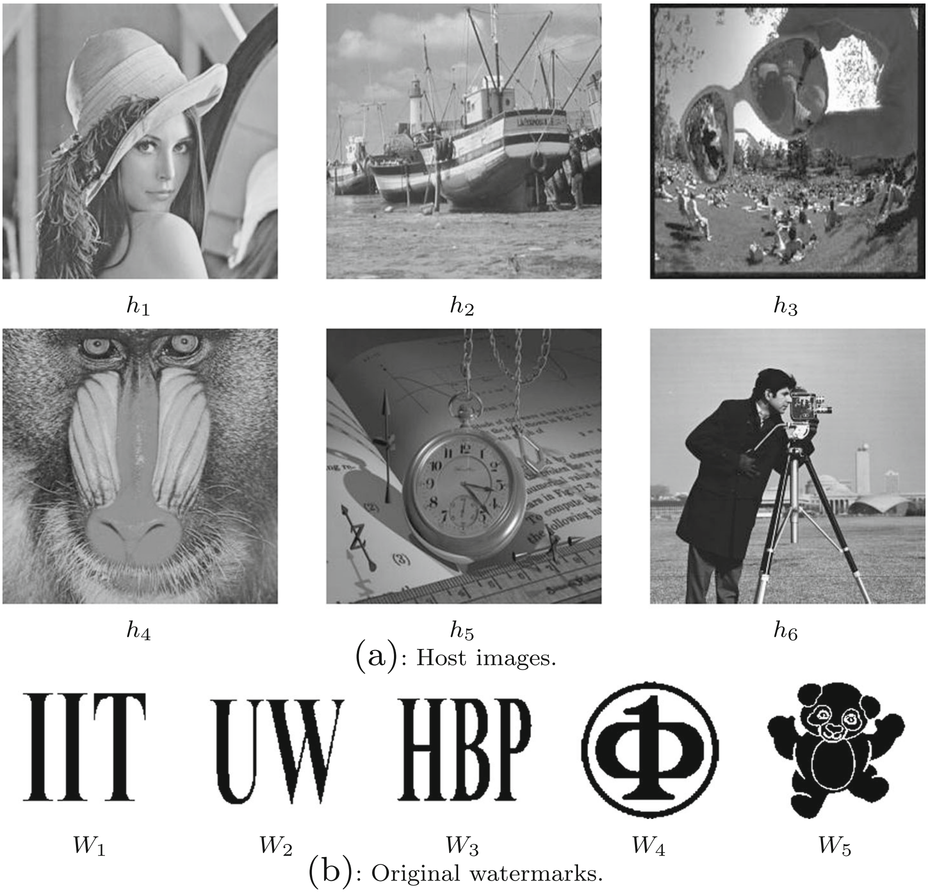
Extraction algorithm

Fig. 10 A numerical example to illustrate the proposed blind watermarking technique in the ROWT domain. In this example, $\alpha = 5$, $\delta = 0.25$, $G(P) = G(N) = 1$ and $k = 1$

4.1 Experiment1: testing of proposed watermarking schemes and effect of embedding strength and error controller

Non-blind watermarking scheme. We have implemented the proposed non-blind watermarking scheme on the data-set (shown in Fig. 11) for various embedding strengths ($\alpha = 1 : 1 : 20$). Peak-signal-to-noise-ratio (PSNR) [30] measures the visual similarity between the original image and the watermarked image, and normalized-Hamming-similarity (NHS) [30] measures the similarity between the embedded watermark and corresponding extracted watermarks. The NHS ranges from zero to one ([0,1]). If NHS is 1, then both watermarks are the same. If NHS is 0, then watermarks are negative of each other. If NHS=0.5, then watermarks are uncorrelated and have no common information. We have observed that the embedding strength decreases PSNR and does not affect NHS. Figure 12a shows a sample watermarked image which corresponds to host image h_4, original watermark W_1, and embedding strength of 3 and ensures no visual degradation in the watermarked image. The obtained watermarked images are 64-bit (double format) gray scale images of size 256 × 256 pixels. Figure 12b, c, and d show watermarks extracted from V, H and D sub-bands, respectively, of a watermarked image (Fig. 12a) and ensure very good quality of extracted watermarks. We have obtained very close results for all other combinations of host images and original watermarks of the data-set (Fig. 11).

Blind watermarking scheme. We have also implemented the proposed blind watermarking scheme on the same data-set for various embedding strengths ($\alpha = 1 : 1 : 20$) and various error controllers ($\delta = 0.00, 0.05, 0.10, 0.15, 0.20, 0.25, 0.30, 0.33$). We have observed that the embedding strength and error controller decrease PSNR and do not



(a): Host images.

(b): Original watermarks.

Fig. 11 Data-set

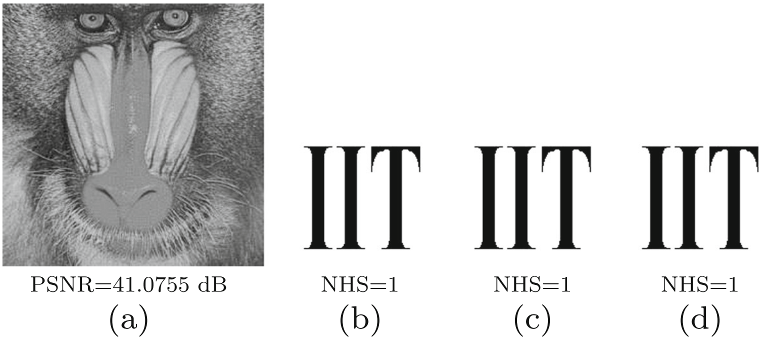


Fig. 12 A sample result of non-blind watermarking scheme at embedding strength of 3. **a**: A sample watermarked image. **b**: Watermark extracted from V sub-bands of (**a**). **c**: Watermark extracted from H sub-bands of (**a**). **d**: Watermark extracted from D sub-bands of (**a**)

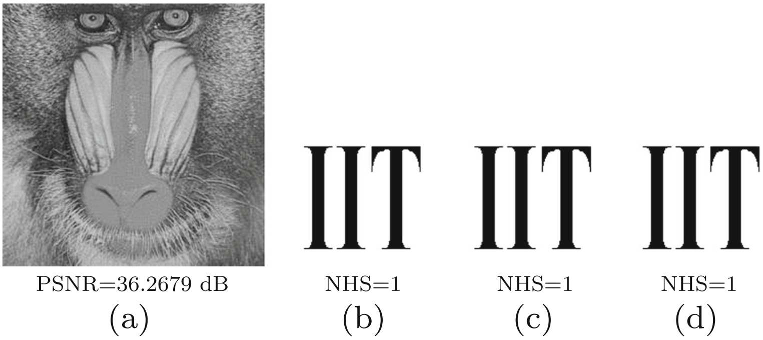


Fig. 13 A sample result of blind watermarking scheme at embedding strength of 5 and error controller of 0.25. **a**: A sample watermarked image. **b**: Watermark extracted from V sub-bands of (**a**). **c**: Watermark extracted from H sub-bands of (**a**). **d**: Watermark extracted from D sub-bands of (**a**)

affect NHS. Figure 13a shows a sample watermarked image which corresponds to host image h_4, original watermark W_1, embedding strength of 5 and error controller δ of 0.25 and ensures no visual degradation in the watermarked image. Like the non-blind watermarking scheme, the obtained watermarked images are 64-bit (double format) gray scale images of size 256 × 256 pixels. Figure 13b, c, d show watermarks extracted from V, H and D sub-bands, respectively, of a watermarked image (Fig. 13a) and ensure very good quality of extracted watermarks. We have obtained very close results for all other combinations of host images and original watermarks of the data-set (Fig. 11).

4.2 Experiment 2: what if the observed property of the ROWT is not utilized?

This experiment implements the ROWT based traditional non-blind and blind watermarking schemes which do not utilize the observed property of the ROWT. For this traditional non-blind watermarking scheme, the fundamental embedding and extraction rules are according to Algorithms 3 and 4, respectively, and for this traditional blind watermarking scheme, the

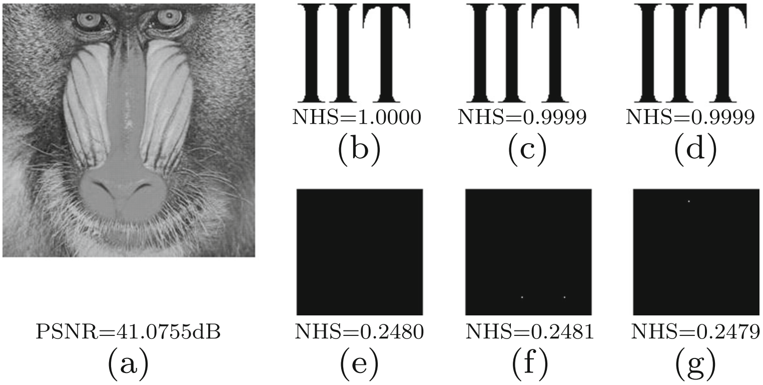


Fig. 14 A traditional non-blind watermarking scheme in the ROWT domain without the observed property. **a**: A sample watermarked image. **b**: Watermark extracted from positive V sub-band of (**a**). **c**: Watermark extracted from positive H sub-band of (**a**). **d**: Watermark extracted from positive D sub-band of (**a**). **e**: Watermark extracted from negative V sub-band of (**a**). **f**: Watermark extracted from negative H sub-band of (**a**). **g**: Watermark extracted from negative D sub-band of (**a**).

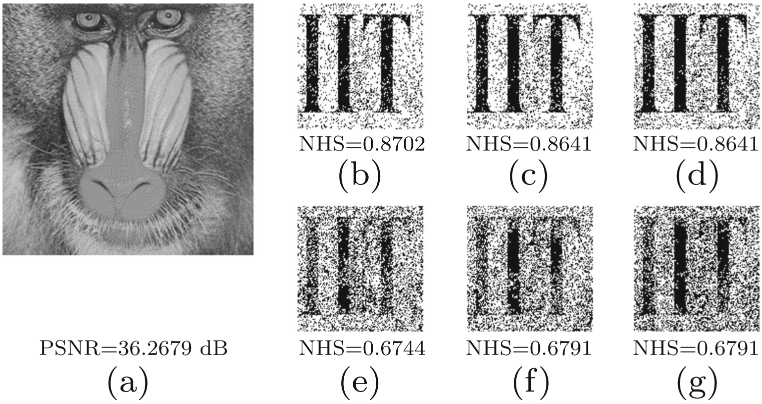


Fig. 15 A traditional non-blind watermarking scheme in the ROWT domain without the observed property. **a**: A sample watermarked image. **b**: Watermark extracted from positive V sub-band of (**a**). **c**: Watermark extracted from positive H sub-band of (**a**). **d**: Watermark extracted from positive D sub-band of (**a**). **e**: Watermark extracted from negative V sub-band of (**a**). **f**: Watermark extracted from negative H sub-band of (**a**). **g**: Watermark extracted from negative D sub-band of (**a**)

fundamental embedding and extraction rules are according to Algorithms 7 and 8, respectively. If we do not incorporate the observed property of the ROWT, then the length of a watermark can be doubled as a watermark can be embedded and extracted from each of the six detailed sub-bands of an image.

Non-blind watermarking scheme. Figure 14 shows a sample result of a traditional non-blind watermarking scheme in the ROWT domain which does not use the observed property of the ROWT. We have observed that watermarks extracted from positive

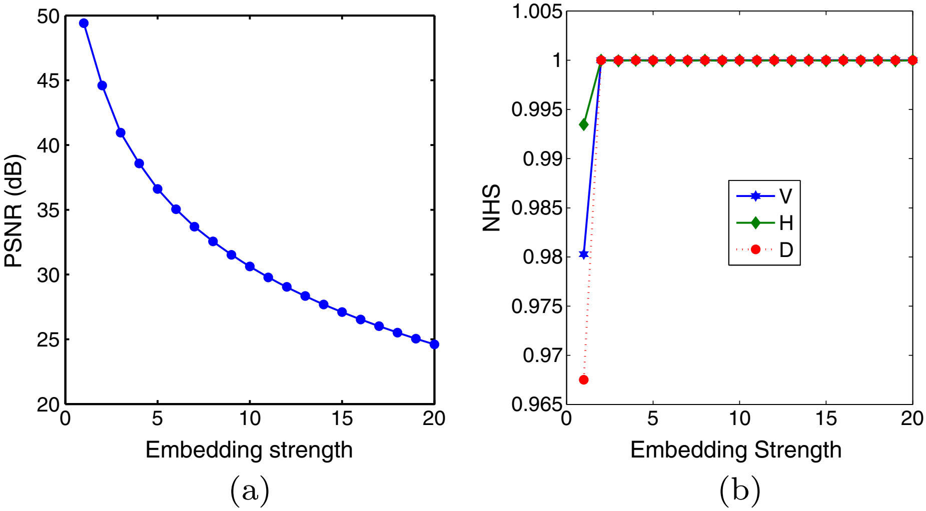


Fig. 16 Non-blind watermarking scheme for a sample combination of host image h_4 and original watermark W_1. **a**: PSNR curve of formatted watermarked images of the combination. **b**: NHS curves of watermarks extracted from V, H and D sub-bands of formatted watermarked images of the combination

sub-bands are very close to embedded watermark and watermarks extracted from negative sub-bands are not matched with embedded watermark. Therefore, only positive sub-bands can be used for watermarking. The slightly better quality of extracted watermarks (see Fig. 12b, c, d and Fig. 14b, c, d) affirms that the proposed non-blind watermarking scheme with the observed ROWT property has slightly better performance than the ROWT based traditional non-blind watermarking scheme.

Blind watermarking scheme. Figure 15 shows a sample result of a traditional blind watermarking scheme in the ROWT domain which does not utilize the observed ROWT property. We have observed that extracted watermarks are very noisy (Fig. 15).

Table 3 Performance of proposed non-blind watermarking scheme in the ROWT domain after formatting attack. $\alpha = 3$

Original	Watermark	PSNR	NHS of extracted watermark		
Image		dB	V sub-band	H sub-band	D sub-band
h_1	W_1	40.96	1	1	1
h_1	W_2	40.84	1	1	1
h_1	W_3	41.41	1	1	1
h_1	W_4	40.98	1	1	1
h_1	W_5	40.88	1	1	1
h_2	W_1	40.96	0.9999	0.9999	0.9999
h_2	W_2	40.84	0.9999	0.9999	0.9999
h_2	W_3	41.40	1	1	1
h_2	W_4	40.98	0.9999	0.9999	0.9999
h_2	W_5	40.88	1	1	1
h_3	W_1	41.04	0.9886	0.9901	0.9955
h_3	W_2	40.93	0.9868	0.9882	0.9937
h_3	W_3	41.50	0.9883	0.9893	0.9941
h_3	W_4	40.97	0.9865	0.9885	0.9938
h_3	W_5	40.85	0.9884	0.9894	0.9949
h_4	W_1	40.96	1	0.9999	1
h_4	W_2	40.84	1	0.9999	1
h_4	W_3	41.40	1	0.9999	1
h_4	W_4	40.97	1	0.9999	1
h_4	W_5	40.89	1	0.9999	1
h_5	W_1	40.96	0.9996	0.9998	0.9999
h_5	W_2	40.84	0.9996	0.9997	0.9998
h_5	W_3	41.41	0.9997	0.9998	0.9999
h_5	W_4	40.98	0.9997	0.9997	0.9998
h_5	W_5	40.88	0.9997	0.9998	0.9999
h_6	W_1	40.96	1	1	1
h_6	W_2	40.84	1	1	1
h_6	W_3	41.40	1	1	1
h_6	W_4	40.97	1	1	1
h_6	W_5	41.96	1	1	1

Figures 13 and 15 ensure that proper utilization of the observed ROWT property has significantly improved the quality of extracted watermarks.

4.3 Experiment 3: the effect of embedding strength and error controller on formatted watermarked images

In Experiment1, host images are eight bit gray scale images and watermarked images are 64-bit (double format) gray scale images. In most of the watermarking applications, the same format of host images and watermarked images is a common request. We have applied the uint8 operation on all the watermarked images to obtain formatted watermarked images (eight bit gray scale images).

Non-blind watermarking scheme. Figure 16a shows the PSNR curve of the sample formatted watermarked images and Fig. 16b shows the NHS curves of the watermarks extracted from the V, H and D sub-bands of the sample formatted watermarked images. The sample formatted watermarked images corresponds to the combination of host image h_4 and watermark W_1. Figure 16a depicts a decrease in the PSNR with embedding strength. Moreover, the PSNR curve is very close to that obtained in Experiment1. Figure 16b depicts that NHSs depend on low range embedding strength and near independence of embedding strength after a value of two. These ensure a slight effect of the uint8 operation on the quality of extracted watermarks in the lower range of embedding strength. Table 3 gives quantitative results of the data-set for each watermarked image at embedding strength of 3. Figure 17a shows a sample formatted watermarked image corresponds to host image h_4, original watermark W_1 and embedding strength of 3, and ensures no visual degradation in the watermarked image. In other experiments, we have used embedding strength of 3, unless stated. Figure 17 b, c, d show watermarks extracted from V, H and D sub-bands, respectively, of the formatted watermarked image (Fig. 11a) and ensure very good quality of extracted watermarks. In summary, the uint8 operation slightly affects the performance of a non-blind watermarking scheme. We have obtained very close results for all other combinations of host images and original watermarks of the data-set (Fig. 11).

Blind watermarking scheme. Figure 18a shows the PSNR curves of sample formatted watermarked images and Fig. 18b, c, d show the NHS curves of watermarks extracted

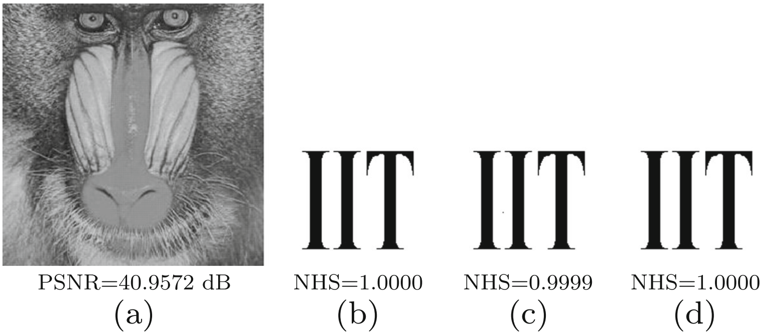


Fig. 17 A sample result of the proposed non-blind watermarking scheme for a formatted watermarked image at an embedding strength of 3. **a**: A sample formatted watermarked image. **b**: Watermark extracted from V sub-bands of (**a**). **c**: Watermark extracted from H sub-bands of (**a**). **d**: Watermark extracted from D sub-bands of (**a**).

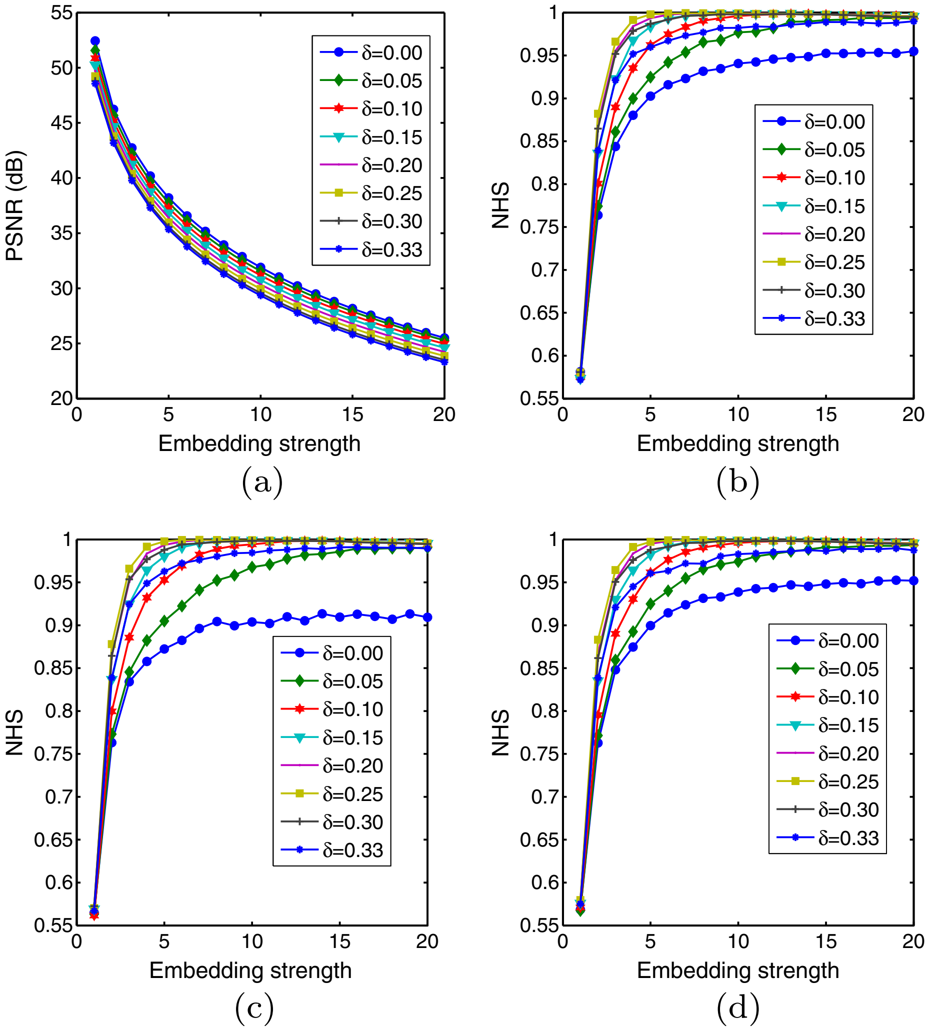


Fig. 18 Blind watermarking scheme for a sample combination of host image h_4 and original watermark W_1. δ is the error controller. **a**: PSNR curves of formatted watermarked images of the combination. **b**: NHS curves of watermarks extracted from V sub-bands of formatted watermarked images of the combination for different error controllers (δs). **c**: NHS curves of watermarks extracted from H sub-bands. **d**: NHS curves of watermarks extracted from D sub-bands

from V, H, and D sub-bands, respectively, of sample formatted watermarked images. The sample formatted watermarked images correspond to the combination of host image h_4 and watermark W_1. Figure 18a depicts a decrease in the PSNR with embedding strength and error controller. Moreover, PSNR curves are very close to those obtained in Experiment1 for the proposed blind watermarking scheme. Figure 18b, c, d depict that embedding strength and error controller affect NHSs significantly. We have observed that at an embedding strength near 5, NHSs curves are at the maximum. Moreover, NHS curves corresponding to an error controller of 0.25 have dominated. Table 4 gives

Table 4 Performance of proposed blind watermarking scheme in the ROWT domain after formatting attack. $\delta = 0.25, \alpha = 5$

Original	Watermark	PSNR	NHS of extracted watermark		
Image		dB	V sub-band	H sub-band	D sub-band
h_1	W_1	36.05	0.9979	0.9988	0.9976
h_1	W_2	35.97	0.9979	0.9973	0.9977
h_1	W_3	35.91	0.9976	0.9972	0.9978
h_1	W_4	35.92	0.9975	0.9978	0.9979
h_1	W_5	35.95	0.9978	0.9980	0.9978
h_2	W_1	35.91	0.9982	0.9982	0.9975
h_2	W_2	35.83	0.9982	0.9980	0.9980
h_2	W_3	36.17	0.9980	0.9979	0.9983
h_2	W_4	35.97	0.9981	0.9985	0.9978
h_2	W_5	35.90	0.9982	0.9983	0.9982
h_3	W_1	36.02	0.9583	0.9592	0.9606
h_3	W_2	35.95	0.9582	0.9583	0.9604
h_3	W_3	36.32	0.9574	0.9583	0.9606
h_3	W_4	35.98	0.9581	0.9584	0.9605
h_3	W_5	36.19	0.9577	0.9586	0.9606
h_4	W_1	36.25	0.9977	0.9971	0.9979
h_4	W_2	35.81	0.9955	0.9951	0.9956
h_4	W_3	35.74	0.9954	0.9946	0.9946
h_4	W_4	35.85	0.9958	0.9957	0.9966
h_4	W_5	35.78	0.9962	0.9952	0.9956
h_5	W_1	36.09	0.9957	0.9951	0.9951
h_5	W_2	35.77	0.9850	0.9855	0.9832
h_5	W_3	35.69	0.9851	0.9855	0.9830
h_5	W_4	35.94	0.9856	0.9856	0.9838
h_5	W_5	35.87	0.9853	0.9854	0.9850
h_6	W_1	36.04	0.9854	0.9851	0.9849
h_6	W_2	35.69	0.9851	0.9855	0.9830
h_6	W_3	36.04	0.9854	0.9851	0.9849
h_6	W_4	35.83	0.9852	0.9853	0.9840
h_6	W_5	36.01	0.9853	0.9852	0.9850

quantitative results of the data-set for each watermarked image at embedding strength of 5 and error controller δ of 0.25. Figure 19a shows a formatted watermarked image which corresponds to host image h_4, original watermark W_1, embedding strength of 5 and error controller δ of 0.25, and ensures no visual degradation in the watermarked image. In other experiments, we have used embedding strength of 5 and error controller δ of 0.25, unless stated. Figure 19b, c, d show watermarks extracted from V, H and D sub-bands, respectively, of the formatted watermarked image (Fig. 19a), and show very slight noise in the extracted watermarks. In summary, we have observed that the uint8 operation affects the performance of the blind watermarking scheme and it has very little

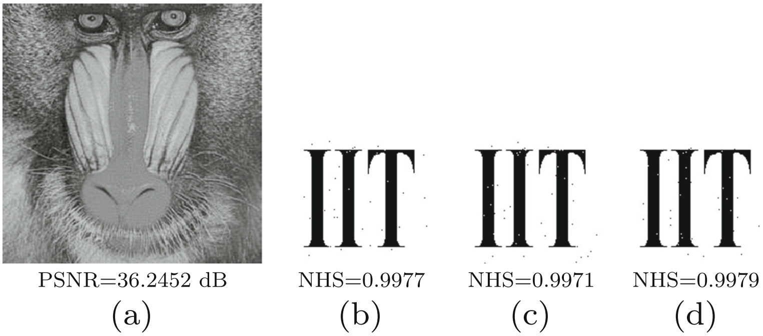


Fig. 19 A sample result of the blind watermarking scheme for a formatted watermarked image at embedding strength of 5 and error controller of 0.25. **a**: A sample formatted watermarked image. **b**: Watermark extracted from V sub-bands of (**a**). **c**: Watermark extracted from H sub-bands of (**a**). **d**: Watermark extracted from D sub-bands of (**a**)

effect when the embedding strength was near a value of 5 and error controller was near a value of 0.25. We have obtained very close results for all other combinations of host images and original watermarks of the data-set (Fig. 11).

4.4 Experiment 4: attack analysis

This experiment evaluates the performance of the proposed non-blind and blind watermarking schemes against various common image processing attacks such as cropping, Gaussian filtering, Gaussian noise and salt and pepper noise. We have applied all underlying attacks on formatted watermarked images of each combination of host images and original watermarks of the data-set (Fig. 11).

Cropping. In cropping from the center, we have blackened a certain percentage of pixels from the center of the formatted watermarked images. We have used the cropping percentage = 0 : 10 : 60. Figure 20a and b show the NHS curves of watermarks extracted from V, H and D sub-bands of sample cropped formatted watermarked images for non-blind and blind watermarking schemes, respectively. Figure 20a and b correspond to the combination of host image h_4 and original watermark W_1. Figure 20a and b depict that an increase in the cropping percentage degrades quality of extracted watermarks for both the non-blind and the blind watermarking schemes. We have obtained very close results for all other formatted watermarked images.

Gaussian filtering. This experiment applies a Gaussian filter of window size 3 × 3 and of different variance on the formatted watermarked images. We have used variance = 0.1 : 0.1 : 1.0. Figure 21a and b show the NHS curves of watermarks extracted from V, H and D sub-bands of sample Gaussian filtered formatted watermarked images for non-blind and blind watermarking schemes, respectively. Figure 21a and b correspond to the combination of host image h_4 and original watermark W_1. Figure 21a and b depict that after a variance of 0.3, the quality of extracted watermarks is significantly degraded for both non-blind and blind watermarking schemes. We have obtained very close results for all other formatted watermarked images.

Gaussian noise. This experiment adds Gaussian noise of zero mean and of different variance in all the formatted watermarked images. We have used Gaussian noise variance =

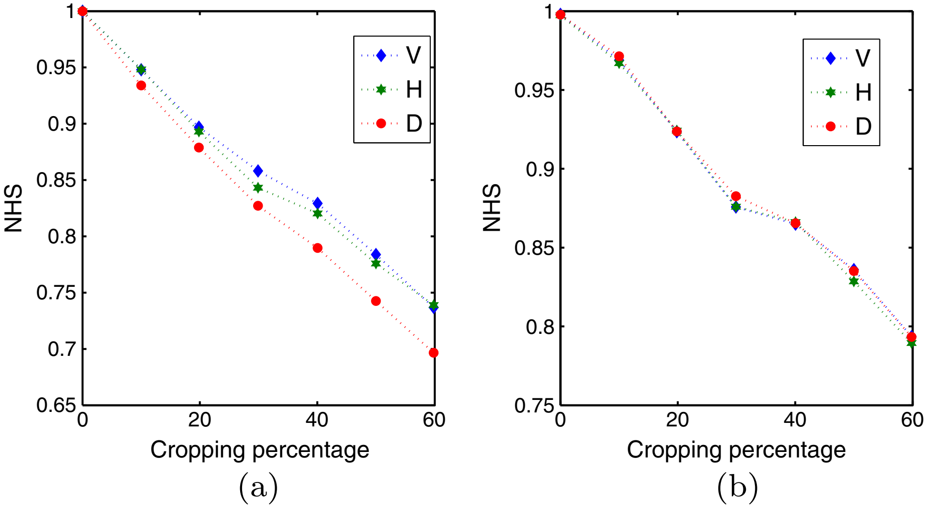


Fig. 20 NHS curves after cropping of formatted watermarked images for combination of host image h_4 and watermark W_1. **a**: Non-blind watermarking scheme. Embedding strength=3. **b**: Blind watermarking scheme. Embedding strength=5, error controller=0.25

$10^{-5} : 10^{-5} : 10^{-4}$. Figure 22a and b show the NHS curves of watermarks extracted from V, H and D sub-bands of sample Gaussian noised formatted watermarked images for non-blind and blind watermarking schemes, respectively. Figure 22a and b correspond to the combination of host image h_4 and original watermark W_1. Figure 22a and b depict that noise variance degrades the quality of extracted watermarks. We have obtained very close results for all other formatted watermarked images.

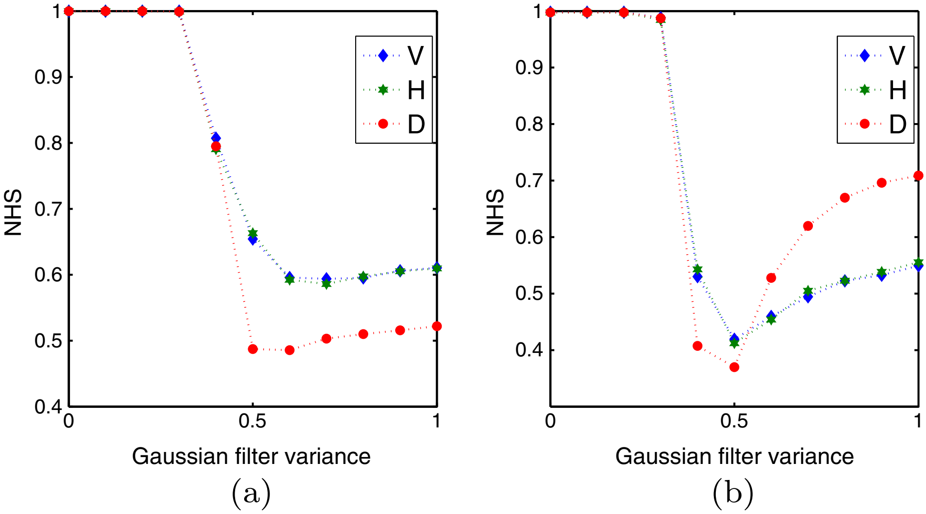


Fig. 21 NHS curves after Gaussian filtering of formatted watermarked images for the combination of host image h_4 and watermark W_1. **a**: Non-blind watermarking scheme. Embedding strength=3. **b**: Blind watermarking scheme. Embedding strength=5, error controller=0.25

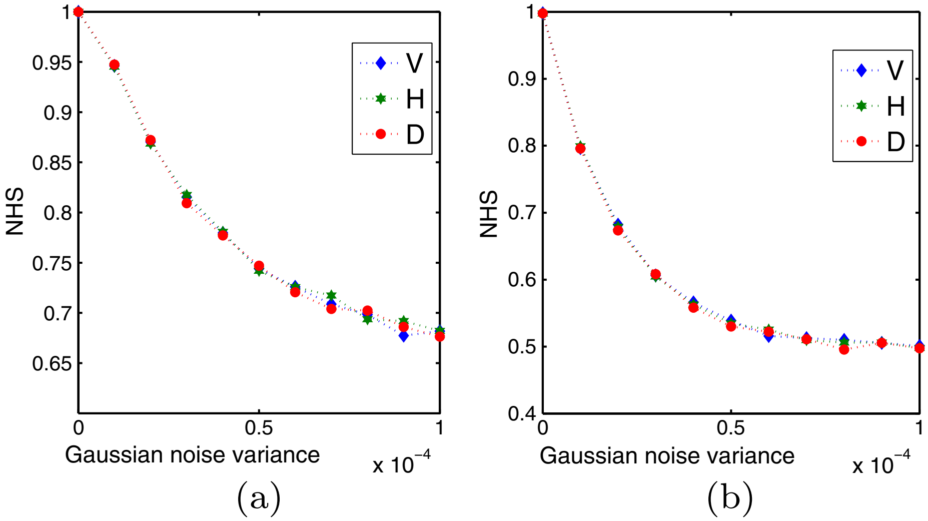


Fig. 22 NHS curves after adding Gaussian noise in formatted watermarked images for combination of host image h_4 and watermark W_1. **a**: Non-blind watermarking scheme. Embedding strength= 3. **b**: Blind watermarking scheme. Embedding strength= 5, error controller= 0.25

Salt and pepper noise. This experiment mixes salt and pepper noise of different density in all the formatted watermarked images. We have used salt and pepper noise density = 0.05 : 0.05 : 1.0. Figure 23a and b show the NHS curves of watermarks extracted from V, H and D sub-bands of sample salt and pepper noised formatted watermarked images for non-blind and blind watermarking schemes, respectively. Figure 23a and b correspond to

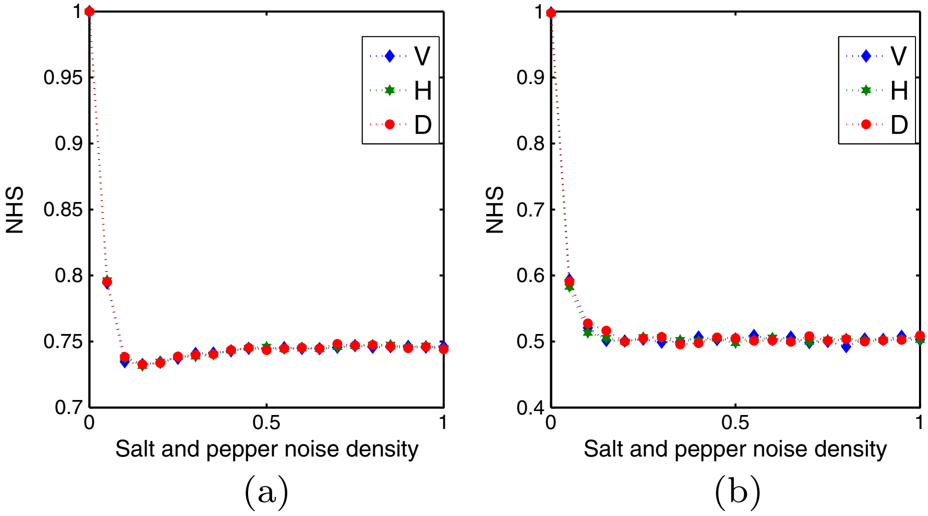


Fig. 23 NHS curves after adding salt and pepper noise in formatted watermarked images for combination of host image h_4 and watermark W_1. **a**: Non-blind watermarking scheme. Embedding strength= 3. **b**: Blind watermarking scheme. Embedding strength= 5, error controller= 0.25

the combination of host image h_4 and original watermark W_1. Figure 23a and b depict that addition of the noise significantly degrades the quality of extracted watermarks. We have obtained very close results for all other formatted watermarked images.

4.5 Experiment 5: comparison without any attack

This experiment compares different related watermarking schemes without any attack. For comparison, we have considered host images of size 256 × 256 pixels. We have compared the size of the watermarks, the capacity and the type of embedded watermarks. Capacity measures the maximum amount of information that can be reliably hidden/extracted. Higher capacity guarantees more information. Capacity and NHS play the same role if the lengths of the watermarks are equal. However, higher NHS does not always guarantee more information (small sized watermarks can have higher NHS but less information). Therefore, we have used capacity to measure the amount of hidden/extracted information.

Ramkumar et al. [29] modeled watermarking as a communication channel and defined the capacity. Figure 24 explains watermarking as a communication channel. In Fig. 24, W_o is an original watermark, W_e is an extracted watermark and $\mathcal{W}$ is the watermarking channel. $\mathcal{W}$ consists of four components: a host image I_o, a watermark embedding algorithm $\oplus$ that embeds W_o in I_o and outputs I_w, a watermark extraction algorithm $\ominus$ that extracts W_e from I_w, and watermarking extraction algorithm noise $n = W_e - W_o$. The capacity per host image is defined as

$$C = M_2 \times N_2 \times \max\{h(W_e) - h(n)\}, \tag{13}$$

where, h is entropy which is defined as

$$h(W) = \begin{cases} -\mathrm{P}_W(0)\log_2(\mathrm{P}_W(0)) & \text{if } \mathrm{P}_W(0) \text{ and } \mathrm{P}_W(1) \neq 0 \\ -\mathrm{P}_W(1)\log_2(\mathrm{P}_W(1)) & \\ \qquad 0 & \text{if } \mathrm{P}_W(0) \text{ or } \mathrm{P}_W(1) = 0 \end{cases}, \tag{14}$$

W is a binary image (watermark), $M_2 \times N_2$ is the size/length of W and $\mathrm{P}_W(0)$ and $\mathrm{P}_W(1)$ are fractions of symbols 0 and 1, respectively, in W. The unit of capacity is bits per host image.

Table 5 gives a comparison between different related watermarking schemes. From Table 5, we observe that the length of the watermarks and the capacity in the proposed watermarking schemes are much higher than in the existing watermarking schemes. The proposed non-blind and blind watermarking schemes have same watermark length and equal capacity. In the proposed watermarking schemes, meaningful binary logos have been used as watermarks. Note that in this experiment, we have considered all combinations of host images and original watermarks of the data-set (Fig. 11) for the proposed non-blind and blind watermarking schemes.

Fig. 24 Watermarking as a communication channel

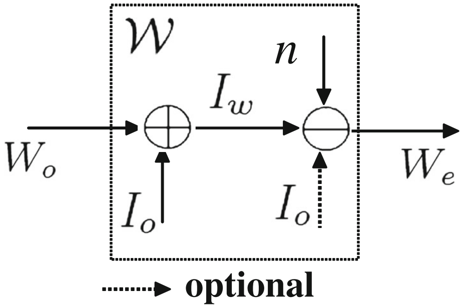


Table 5 Comparison of different watermarking schemes for host images of size 256 × 256 pixels. logo: meaningful binary logo, random: random binary sequence

Scheme	Domain	Length of watermark (pixels)	Capacity (bits)	Watermarks
Blind*	ROWT	$3 \times 128 \times 128$	44,410	logo
Non-blind*	ROWT	$3 \times 128 \times 128$	44,410	logo
Loo [22]	DTCWT	–	5,300	random
Terzija [39]	DTCWT	180	≤ 180	random
Coria [4]	DTCWT	1,024	$\leq 1,024$	random
LFC-DE [43]	DTCWT	200	≤ 200	random
GHFC-DE [43]	DTCWT	700	≤ 700	random

*Proposed schemes

4.6 Experiment 6: robustness comparison

This experiment compares the robustness of different related watermarking schemes against various common attacks. We have used capacity to compare the robustness. Higher capacity ensures higher robustness. In this experiment, we have evaluated the capacity of the proposed non-blind and blind watermarking schemes after corresponding attacks on watermarked images. Note that the capacity evaluation for the proposed watermarking schemes considers all combinations of host images and watermarks of the data-set (Fig. 11) at the points of evaluation. We have reported the capacity of other watermarking schemes without any attack.

Formatted watermarked images. Figure 25 gives the capacities of the proposed watermarking schemes after a formatting attack on the watermarked images. The proposed non-blind watermarking scheme has the highest robustness, very close to the robustness of the proposed blind watermarking scheme. Other watermarking schemes have very low robustness. Certainly, after any kind of attack on the watermarked images, the capacity of other watermarking schemes will be degraded.

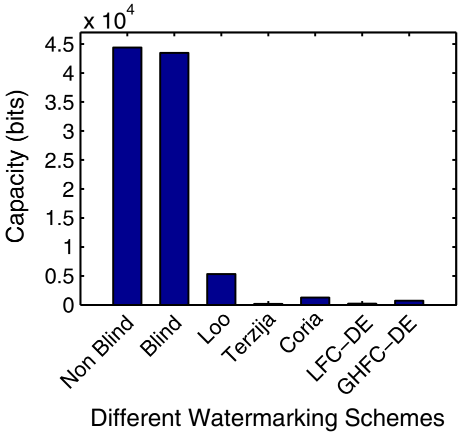
Cropping. Figure 26 gives the capacities of the proposed watermarking schemes after cropping the formatted watermarked images for different cropping percentages. Figure 26 depicts the following:

- Increase in cropping percentage decreases capacities of non-blind and blind watermarking schemes.
- Non-blind watermarking scheme has the highest robustness.
- Blind watermarking scheme has the second highest robustness up to 30 percent cropping.

Gaussian filtering. Figure 27 gives the capacities of the proposed watermarking schemes after Gaussian filtering the formatted watermarked images for different filter variances. Figure 27 depicts the following:

- Filter variance does not affect the capacities of the proposed non-blind and blind watermarking schemes up to a filter variance of 0.3. After that, the filter variance drastically decreases the capacities of both watermarking schemes.

Fig. 25 Robustness comparison of different watermarking schemes after a formatting attack on the watermarked images. Capacities of the proposed non-blind and blind watermarking schemes are evaluated for the data-set (Fig. 11)



- Non-blind watermarking scheme has the highest robustness up to a filter variance of 0.5.
- Blind watermarking scheme has the second best robustness up to a filter variance of 0.5.
- Filter variance saturates non-blind watermarking scheme capacity after a filter variance of 0.6. Moreover, the capacity of the non-blind watermarking scheme converges to the reported capacity of the Terzija et al. [39] scheme, Coria et al. [4] scheme, and LFC-DE and GHFC-DE [43] schemes after a filter variance of 0.6.

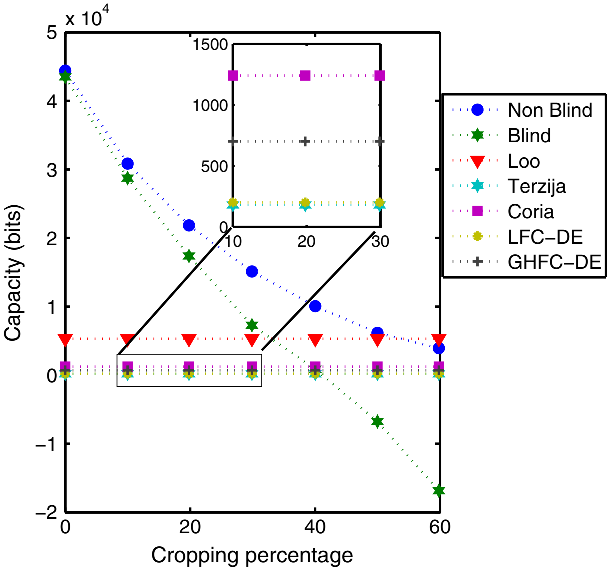


Fig. 26 Robustness comparison of different watermarking schemes after cropping the formatted watermarked images. Capacities of the proposed non-blind and blind watermarking schemes are evaluated for the data-set (Fig. 11)

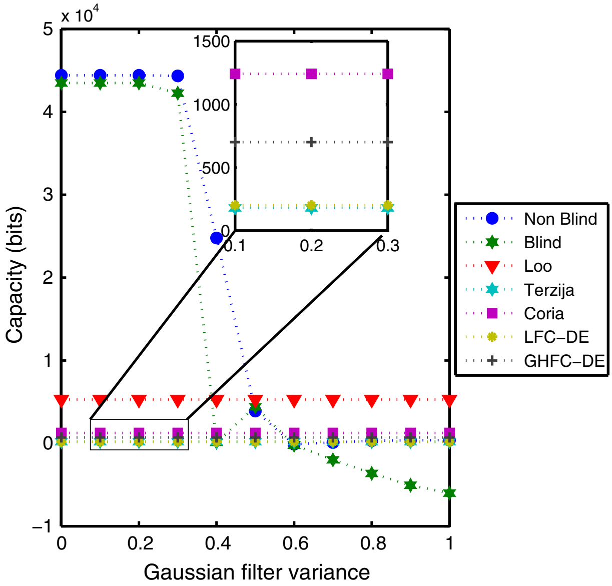


Fig. 27 Robustness comparison of different watermarking schemes after Gaussian filtering the formatted watermarked images. Capacities of the proposed non-blind and blind watermarking schemes are evaluated for the data-set (Fig. 11)

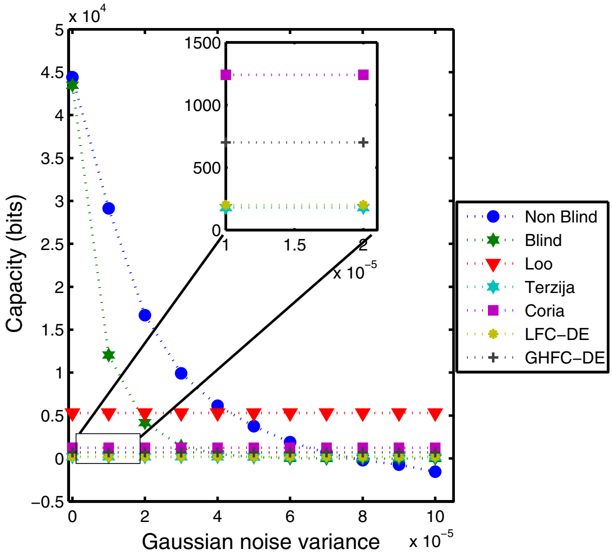


Fig. 28 Robustness comparison of different watermarking schemes after adding Gaussian noise in the formatted watermarked images. Capacities of the proposed non-blind and blind watermarking schemes are evaluated for the data-set (Fig. 11)

Gaussian noise. Figure 28 gives the capacities of the proposed watermarking schemes after adding Gaussian noise in the formatted watermarked images for different noise variances. Figure 28 depicts the following.

- Increase in noise variance decreases the capacities of non-blind and blind watermarking schemes.
- Non-blind watermarking scheme has the best robustness up to a noise variance of 5×10^{-5}.
- Blind watermarking scheme has the second best robustness up to a noise variance of 2×10^{-5}.
- Capacity of the blind watermarking scheme converges to the reported capacity of the Terzija et al. [39] scheme, Coria et al. [4] scheme, and LFC-DE and GHFC-DE [43] schemes, after a noise variance of 3×10^{-5}.

Salt and pepper noise. Figure 29 gives the capacities of the proposed watermarking schemes after adding salt and pepper noise in the formatted watermarked images for different noise densities. Figure 29 depicts the following:

- Increase in noise variance drastically decreases the capacities of non-blind and blind watermarking schemes.
- Non-blind and blind watermarking schemes have the best robustness up to noise density of 0.05.
- Non-blind watermarking scheme has the worst robustness from a noise density of 0.1.
- Capacity of the blind watermarking scheme converges to the reported capacity of Terzija et al. [39] scheme, Coria et al. [4] scheme, and LFC-DE and GHFC-DE [43] schemes, after a noise density of 0.1.

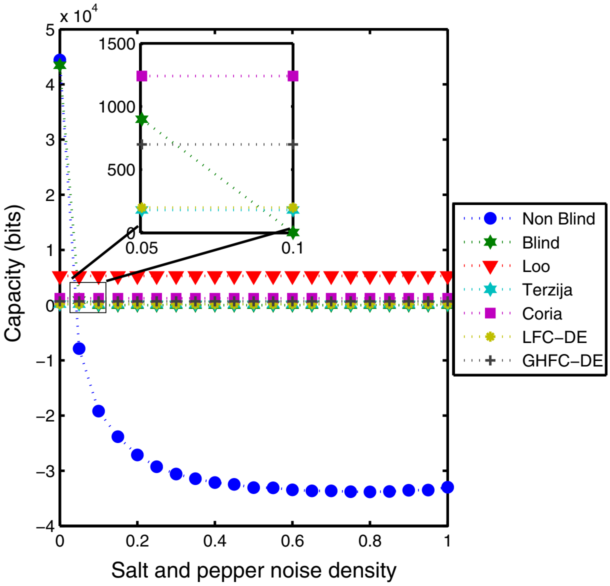


Fig. 29 Robustness comparison of different watermarking schemes after adding salt and pepper noise in the formatted watermarked images. Capacities of the proposed non-blind and blind watermarking schemes are evaluated for the data-set (Fig. 11)

5 Conclusion

Non-blind and blind watermarking schemes have been developed in the ROWT domain. The proposed watermarking schemes have improved capacity and improved watermark length. The drastic improvement in the capacity and the length of the watermarks is achieved due to the observed property of the ROWT. Experiments are conducted on a data-set of six gray scale images and five meaningful binary logo watermarks. Experimental results reveal the following:

- The capacity of the proposed non-blind watermarking scheme and the visual quality of the corresponding watermarked images change with the embedding strength. We have observed that the embedding strength of a value near 3 is optimum for formatted watermarked images. Near optimum embedding strength, the capacity is at its maximum and formatted watermarked images are visually undistinguishable from corresponding original images.
- The performance of the blind watermarking scheme can be controlled by the embedding strength and the error controller. We have observed that the embedding strength and error controller pair is optimum near tuple of (5,0.25) for formatted watermarked images. Near an optimum pair, capacity is at its maximum and formatted watermarked images are visually undistinguishable from corresponding original images.
- Positive sub-bands can be used for a traditional non-blind watermarking scheme as extracted watermarks from positive sub-bands are very close to original watermarks. However, for a traditional blind watermarking scheme, extracted watermarks from all sub-bands are very noisy. Therefore, the observed property of the ROWT is significant for blind watermarking scheme and is slightly important for the non-blind watermarking scheme.
- We have tested both proposed watermarking schemes against various attacks such as cropping, Gaussian filter, Gaussian noise and salt and pepper noise. We have observed that the proposed non-blind and blind watermarking schemes have better robustness than existing DTCWT based watermarking schemes. Moreover, the proposed non-blind watermarking scheme has better robustness than the proposed blind watermarking scheme.

Acknowledgment One of the authors, Himanshu Agarwal, acknowledges the University Grants Commission (UGC) of New Delhi, India for granting him a scholarship under the JRF scheme for his research. He also acknowledges the grant from the Canadian Bureau for International Education under the Canadian Commonwealth Scholarship Program.

References

1. Agarwal H, Raman B, Venkat I (2014) Blind reliable invisible watermarking method in wavelet domain for face image watermark. Multimedia Tools and Applications. doi:10.1007/s11042-014-1934-1
2. Bhatnagar G, Raman B (2011) A new robust reference logo watermarking scheme. Multimedia Tools and Applications 52(2- 3):621–640
3. Bhatnagar G, Wu QJ (2013) Biometrics inspired watermarking based on a fractional dual tree complex wavelet transform. Futur Gener Comput Syst 29(1):182–195

4. Coria LE, Pickering MR, Nasiopoulos P, Ward RK (2008) A video watermarking scheme based on the dual-tree complex wavelet transform. IEEE Transactions on Information Forensics and Security 3(3):466–474
5. Cox I, Miller M, Bloom J, Fridrich J, Kalker T (2007) Digital Watermarking and Steganography, 2nd edn. Morgan Kaufmann Publishers Inc., San Francisco
6. Cox IJ, Kilian J, Leighton FT, Shamoon T (1997) Secure spread spectrum watermarking for multimedia. IEEE Trans Image Process 6(12):1673–1687
7. Fallahpour M (2011) High capacity audio watermarking using the high frequency band of the wavelet domain. Multimedia tools and Applications 52(2-3):485–498
8. Gonde AB, Maheshwari RP Balasubramanian R (2010) Complex wavelet transform with vocabulary tree for content based image retrieval. In: Proceedings of the Seventh Indian Conference on Computer Vision, Graphics and Image Processing, ACM, Chennai
9. Gonzalez RC, Woods RE (2008) Digital Image Processing. Prentice Hall
10. Holliman M, Memon N (2000) Counterfeiting attacks on oblivious block-wise independent invisible watermarking schemes. IEEE Trans Image Process 9(3):432–441
11. Jain AK, Uludag U (2003) Hiding biometric data. IEEE Transactions on Pattern Analysis and Machine Intelligence 25(11):1494–1498
12. Khana A, Malika SA, Alib A, Chamlawia R, Hussaina M, Mahmoodc MT, Usmand I (2012) Intelligent reversible watermarking and authentication: Hiding depth map information for 3d cameras. Elsevier, Inf Sci 216:155–175
13. Kim WG, Lee HK (2009) Multimodal biometric image watermarking using two-stage integrity verification. Sig Process 89(12):2385–2399
14. Kokare M, Biswas PK, Chatterji BN (2005) Texture image retrieval using new rotated complex wavelet filters. IEEE Transactions on Systems, Man, and Cybernetics Part B: Cybernetics 35(6):1168–1178
15. Kokare M, Biswas PK, Chatterji BN (2006) Rotation-invariant texture image retrieval using rotated complex wavelet filters. IEEE Transactions on Systems, Man, and Cybernetics Part B: Cybernetics 36(6):1273–1282
16. Korus P, Białas J, Dziech A (2012) A new approach to high-capacity annotation watermarking based on digital fountain codes. Multimedia Tools and Applications:1–19
17. Kumar S, Kumar S, Raman B, Sukavanam N (2013) Image disparity estimation based on fractional dual-tree complex wavelet transform: A multi-scale approach. International Journal of Wavelets. Multiresolution and Information Processing 11(1):1350,004,1–1350,004,21
18. Kundur D, Hatzinakos D (1999) Digital watermarking for telltale tamper proofing and authentication. Proceedings of IEEE 87(7):1167–1179
19. Kundur D, Hatzinakos D (2004) Toward robust logo watermarking using multiresolution image fusion principles. IEEE Trans Multimed 6(1):185–198
20. Li CT, Yang FM, Lee CS (2002) Oblivious fragile watermarking scheme for image authentication. IEEE International Conference on Acoustics, Speech, and Signal Processing 4:IV–3445 – IV–3448
21. Lina TC, Linb CM (2009) Wavelet-based copyright-protection scheme for digital images based on local features. Elsevier, Inf Sci 179(19):3349–3358
22. Loo P, Kingsbury NG (2000) Digital watermarking using complex wavelets. In: International Conference Image Processing, IEEE, Vancouver, vol 3
23. Ma B, Wang Y, Li C, Zhang Z, Huang D (2013) Secure multimodal biometric authentication with wavelet quantization based fingerprint watermarking. Multimedia Tools and Applications. doi:10.1007/s11042-013-1372-5
24. Magarey J, Kingsbury NG (1998) Motion estimation using a complex-valued wavelet transform. IEEE Transactions on Signal Processing 46(4):1069–1084
25. Mallat SG (1989) A theory for multiresolution signal decomposition: The wavelet representation. IEEE Transactions on Pattern Analysis and Machine Intelligence 2(7):674–693
26. Miller M, Kingsbury N, Hobbs R (2005) Seismic imaging using complex wavelets. In: Proceeding of International Conference on Acoustic, Speech, Signal Processing (ICASSP), IEEE, Philadelphia, vol 2
27. Mohanty SP (2009) A secure digital camera architecture for integrated real-time digital rights management. J Syst Archit 55(10):468–480
28. Mohanty SP, Ramakrishnan KR, Kankanhalli MS (2000) A dct domain visible watermarking technique for images. IEEE International Conference on Multimedia and Expo 2:1029–1032
29. Ramkumar M, Akansu AN (1998) Information theortic bounds for data hiding in compressed images. In: IEEE Second Workshop on Multimedia Signal Processing

30. Rani A, Raman B, Kumar S (2013) A robust watermarking scheme exploiting balanced neural tree for rightful ownership protection. Multimedia Tools and Applications:1–24
31. Romberg J, Choi H, Baraniuk R, Kingsbury N (2000) Multiscale classification using complex wavelets and hidden markov tree models. In: Proceeding International Conference Image Processing, Vancouver, vol 2
32. Selesnick I (2014) 2-d dual-tree wavelet transform. http://eeweb.poly.edu/iselesni/WaveletSoftware/dt2D.html, last accessed on April, 2014
33. Selesnick IW, Baraniuk RG, Kingsbury NG (2005) The dual-tree complex wavelet transform. IEEE Signal Procesing Magazine:123–151
34. Shaffrey CW, Kingsbury NG, Jermyn IH (2002) Unsupervised image segmentation via markov trees and complex wavelets. In: Proceedings of Internationl Conference on ImageProcessing, IEEE, Rochester, vol 3
35. Shia X, Xiaoa D (2013) A reversible watermarking authentication scheme for wireless sensor networks. Elsevier, Inform Sci 240:173–183
36. Skodras A, Christopoulos C, Ebrahimi T (2001) The jpeg 2000 still image compression standard. IEEE Signal Procesing Magazine:36–58
37. Suhail M, Obaidat M, Ipson S, Sadoun B (2003) A comparative study of digital watermarking in jpeg and jpeg 2000 environments. Elsevier, Inform Sci 15:93–105
38. Tang CW, Hang HM (2003) A feature-based robust digital image watermarking scheme. IEEE Transactions on Signal Processing 51(4):950–959
39. Terzija N, Geisselhardt W (2004) Digital image watermarking using complex wavelet transform. In: Proceeding Workshop Multimedia Security, ACM, Magdeburg
40. Wallace GK (1992) The jpeg still picture compression standard. IEEE Trans Consum Electron 38(1):18–34
41. Wang N, Men C (2013) Reversible fragile watermarking for locating tampered blocks in 2d vector maps. Multimedia Tools and Applications 67(3):709–739
42. Wong PW, Memon N (2001) Secret and public key image watermarking schemes for image authentication and ownership verification. IEEE Trans Image Process 10(10):1593–1601
43. Yang H, Jiang X, Kot AC (2011) Embedding binary watermarks in dual-tree complex wavelets domain for access control of digital images. Springer, Transactions on DHMS 6:18–36
44. Ye Z, Lu CC (2003) A complex wavelet domain markov model for image denoising. In: Proceedings of International Conference on Image Processing, IEEE, Barcelona, vol 3

Himanshu Agarwal is a full time Ph.D. student in the Department of Mathematics, Indian Institute of Technology Roorkee since July 2009. He is a member of Computer Vision Graphics and Image Processing Laboratory in the Department of Mathematics, Indian Institute of Technology Roorkee since July 2009. He was a visiting research student in the Applied Computer Science Department, The University of Winnipeg, Canada in 2012–2013 for six months. He has received M.Sc. degree in Industrial Mathematics & Informatics from Indian Institute of Technology Roorkee in 2009. So far he has published five research papers in reputed International Journals and International conferences. His area of research includes digital watermarking, biometrics and applications of wavelets in images.

Pradeep K. Atrey is an Assistant Professor in the Department of Computer Science, College of Computing and Information, State University of New York at Albany, USA. He is also an Associate Professor (on leave) in Department of Applied Computer Science at the University of Winnipeg, Canada. He received his Ph.D. in Computer Science from the National University of Singapore, M.S. in Software Systems and B.Tech. in Computer Science and Engineering from India. He was a Postdoctoral Researcher at the Multimedia Communications Research Laboratory, University of Ottawa, Canada. His current research interests are in the area of Multimedia Computing with a focus on Multimedia Surveillance and Privacy, Image/Video Security, and Social Media. He has authored/co-authored over 73 research articles at reputed ACM, IEEE, and Springer journals and conferences. Dr. Atrey is on the editorial board of several journals including ACM Trans. on Multimedia Computing, Communications and Applications and ETRI Journal. He has been associated with over 20 international conferences in various roles such as General Chair, Program Chair, Publicity Chair, Web Chair, and TPC Member. Dr. Atrey was a recipient of the Erica and Arnold Rogers Award for Excellence in Research and Scholarship (2014), the ETRI Journal Best Editor Award (2012), ETRI Journal Best Reviewer Award (2009) and the two University of Winnipeg Merit Awards for Exceptional Performance (2010 and 2012). He was also recognized as "ICME 2011Quality Reviewer".

Balasubramanian Raman is an associate professor in the Department of Computer Science and Engineering at Indian Institute of Technology Roorkee from 2013. He has obtained MSc degree in mathematics from Madras Christian College (University of Madras) in 1996 and PhD from Indian Institute of Technology Madras in 2001. He was a post doctoral fellow at University of Missouri Columbia, USA in 2001–2002 and a post doctoral associate at Rutgers, the State University of New Jersey, USA in 2002–2003. He joined Department of Mathematics at Indian Institute of Technology Roorkee as lecturer in 2004 and became assistant professor in 2006 and associate professor in 2012. He was a visiting professor and a member of Computer Vision and Sensing Systems Laboratory in the Department of Electrical and Computer Engineering at University of Windsor, Canada during MayAugust 2009. His area of research includes Vision Geometry, Digital Watermarking using Mathematical Transformations, Image Fusion, Biometrics and Secure Image Transmission over Wireless Channel, Content Based Image Retrieval and Hyperspectral Imaging.