

A HIGH CAPACITY REVERSIBLE WATERMARKING TECHNIQUE BASED ON DIFFERENCE EXPANSION

Stewart I. Fraser

Department of Engineering
University of Aberdeen
Aberdeen, AB24 3UE, UK
Email: s.i.fraser@abdn.ac.uk

Alastair R. Allen

Department of Engineering
University of Aberdeen
Aberdeen, AB24 3UE, UK
Email: a.allen@abdn.ac.uk

ABSTRACT

Reversible image watermarking is a special case of watermarking that can embed/recover a user payload **and** restore the original un-watermarked image **exactly**. Being able to restore the original image is important for modalities such as medical and military imaging. This paper presents a high capacity reversible watermarking scheme based on difference expansion. The novel method presented here improves upon a previous scheme by introducing the companding technique. Experiments compare the novel and previous schemes using test, medical and military images. It can be seen from the results that the novel scheme has a significantly higher payload hiding capacity than the previous scheme.

KEY WORDS

Digital Image Processing, Reversible Watermarking, Medical and Military Imaging.

1 Introduction

In digital image watermarking, a hardly noticeable noise-like signal is embedded into an image to: (i) protect it, (ii) authenticate it or (iii) enrich its information content [1]. The watermarking process usually introduces a degradation (via bit replacement, quantization, *etc.*) into the image. Although this degradation is slight, it is unacceptable in certain applications which have sensitive imagery. Two such applications are medical imaging and military imaging. In medical imaging, access to the unaltered original image is of paramount importance due to legal reasons [2]. In military imaging, images are viewed under non-standard conditions (*e.g.* extreme enhancement and zoom) thus adding a noise like signal in the form of a watermark is unacceptable.

Reversible watermarking is a special case of watermarking with an intriguing feature: once the watermark has been authenticated, it is possible to remove the watermark and recover the the original un-watermarked image **exactly** [3]. This reversal process is achievable without

any reference to information beyond what is in the watermarked image itself. Various methods for reversible watermarking have been proposed, such as schemes that use data compression [2, 4], schemes that use histogram bin exchanging [5, 6, 7] and schemes that use difference expansion [1, 3, 8, 9]. In this paper, a new high capacity reversible watermarking method based on difference expansion is introduced. The performance of this new method is assessed using test, medical and military images.

The novel reversible watermarking method described here is based upon the Tian method [3]; it improves upon the Tian method by introducing a companding technique. This allows the novel scheme to embed a significantly larger payload than is possible with the Tian scheme. Like the Tian method, the novel method can recover the original image **exactly**.

2 Overview of the Tian method

In [3], Tian introduced a reversible watermarking algorithm based upon difference expansion of adjacent pixels. The difference between adjacent pixels was calculated via an integer transform.

2.1 An integer transform

For a pair of pixel values (x, y) in a grayscale image I ($x, y \in \mathbb{Z}; 0 \leq x, y \leq 255$), their average value l is defined as:

$$l = \left\lfloor \frac{x + y}{2} \right\rfloor \quad (1)$$

and thier difference value h is defined as:

$$h = x - y \quad (2)$$

where $\lfloor \cdot \rfloor$ is the floor function. The original x and y values can be obtained from l and h by applying the inverse integer transform described in equations 3 and 4:

$$x = l + \left\lfloor \frac{h + 1}{2} \right\rfloor \quad (3)$$

$$y = l - \left\lfloor \frac{h}{2} \right\rfloor \quad (4)$$

2.2 Watermark embedding via difference expansion

In [3], watermark information is embedded in a digital image by altering the difference value h . There are certain restrictions in the alteration of the h value that must be adhered to. As grayscale values (x, y) in digital images are bounded in the range $[0, 255]$, the inverse integer transform must meet the following two conditions:

$$0 \leq l + \left\lfloor \frac{h+1}{2} \right\rfloor \leq 255 \quad (5)$$

and

$$0 \leq l - \left\lfloor \frac{h}{2} \right\rfloor \leq 255 \quad (6)$$

which is equivalent to:

$$|h| \leq \min(2(255 - l), 2l + 1) \quad (7)$$

Therefore, to prevent underflow and overflow occurring in the watermarked image, the difference value h must satisfy equation 7. Adhering to equation 7, the h value is altered to embed a watermark bit. There are two methods in which to alter h :

1. **Change** the difference value h
2. **Expand** the difference value h

If a neighbouring pixel pair cannot be changed or expanded, they are left unaltered.

2.2.1 Changing the difference value h

For a neighbouring grayscale pair of pixels (x, y) , the difference value h is changeable only if:

$$\left| \left\lfloor \frac{h}{2} \right\rfloor \cdot 2 + b \right| \leq \min(2(255 - l), 2l + 1) \quad (8)$$

for both $b = 0$ and $b = 1$ (b is a watermark bit). Changing h requires removing its LSB (Least Significant Bit) and replacing it with a watermark bit:

$$h_{new} = \left\lfloor \frac{h}{2} \right\rfloor \cdot 2 + b \quad (9)$$

The removed LSB from h must be stored in order to recover the original image exactly after watermark extraction. An exception is made if h is 1 or -2, no LSB is stored. Note that if a pixel pair are identified as changeable and h has a value of 0 or -1, then the pair are left unaltered¹. Thus, changing the value of h does not provide extra storage space.

¹this is because the conditions for changeable and expandable are the same when h is 0 or -1; expanding is preferred over changing

2.2.2 Expanding the difference value h

On the other hand, expanding h does provide extra storage space. Neighbouring grayscale pairs of pixels (x, y) are expandable only if:

$$|2 \cdot h + b| \leq \min(2(255 - l), 2l + 1) \quad (10)$$

for both $b = 0$ and $b = 1$ (b is a watermark bit). Expanding h requires left shifting all the bits by one and using the freed LSB to store the watermark bit:

$$h_{new} = 2 \cdot h + b \quad (11)$$

Thus, if a pair of neighbouring pixels are expandable, one extra bit of storage space is obtained. This is the concept of difference expansion.

It is important to note that an expandable h is also changeable. After embedding a watermark bit via difference expansion, an expanded h_{new} is still changeable. This relationship is exploited in watermark recovery.

In order to reduce the visual degradation to a marked image, a user defined threshold is introduced to expandable pairs. If h_{new} in an expandable pair is greater than or equal to this threshold, the pair are changed rather than expanded, thus minimizing the Mean Squared Error (MSE) of the watermarked image. However, this threshold causes a detection problem [10]. In some situations, the detector cannot tell if a pair has been changed or expanded. For example, suppose that the threshold value is 10, the difference value $h = 6$ and the watermark value to be embedded is 1. From equation 11, $h_{new} = 2 \cdot h + b = 2 \cdot 6 + 1 = 13$. In the retrieving phase at the detector, there are now **two** possibilities: h was 6 (embedded) or h was 13 (not embedded, larger than or equal to threshold). In order to combat this problem, it is necessary to create a binary location map marking the position of all expanded pairs.

2.3 Constructing the watermark bitstream

In Tian's reversible watermarking scheme, the watermark bitstream **W** that is embedded has three distinct components. These are:

- **L**: a binary location map marking the location of expandable pairs. The location map is losslessly compressed² via JBIG compression [11] or run-length coding. The greater the compression of **L**, the greater the space there will be for the payload **P**. An end of message symbol is appended to the end of **L**.

²JBIG lossless compression is used in this paper

- **C**: a vector containing all the removed LSBs from the changeable pairs.
- **P**: the payload the user wishes to embed.

These vectors are concatenated together to produce a watermark bitstream $\mathbf{W} = \mathbf{L} \cup \mathbf{C} \cup \mathbf{P}$. Thus \mathbf{W} contains both the user inserted payload (**P**) and the information required (**L** and **C**) to exactly restore the original image. This is shown diagrammatically in Figure 1.

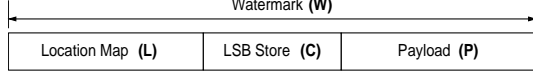


Figure 1. Watermark structure of the Tian scheme

2.4 Watermark recovery

The integer transform (described in section 2.1) is applied to a watermarked image I' . Using equation 8, changeable and expandable pairs are found (**CE'**). Note that, by definition, expandable pairs are also changeable, thus equation 8 finds both changed and expanded pairs.

It can be seen from equations 9 and 11 that the original watermark bitstream \mathbf{W} was inserted into the LSBs of changeable and expandable difference values. Thus, collecting the LSBs from the difference values of **CE'** will result in the recovered watermark bitstream \mathbf{W}' .

By identifying the end of message symbol in \mathbf{W}' , the bits from the start until the end of message symbol are losslessly decompressed (using a JBIG decoder) to retrieve the binary location map. This location map gives the locations of all expanded pairs.

2.4.1 Restoring the original image exactly

The bits in \mathbf{W}' after the location map \mathbf{L}' are the stored LSBs of changeable pairs \mathbf{C}' . Using equations 1 and 2, the integer transform is applied to the watermarked image I' to obtain l' and h' . Using the stored LSBs (c_i) from \mathbf{C}' and the location map values, the original difference values h_o can be computed. For each pixel pair:

- If pixel pair are non changeable, do nothing (**unaltered pair**)
- If pixel pair are changeable and location map is 1 (**expandable pair**)
 $h_o = \lfloor h'/2 \rfloor$

- If pixel pair are changeable and location map is 0 (**changeable pair**)
 - if $(h' == 0)$ or $(h' == 1)$
 $h_o = 1;$
 - else if $(h' == -1)$ or $(h' == -2)$
 $h_o = -2;$
 - else
 $h_o = \lfloor h'/2 \rfloor \cdot 2 + c_i;$
 $i = i + 1;$

Applying the inverse integer transform, the restored image values (x_r and y_r) for each pixel pair can be calculated to obtain the original un-watermarked image **exactly**:

$$x_r = l' + \left\lfloor \frac{h_o + 1}{2} \right\rfloor \quad (12)$$

$$y_r = l' - \left\lfloor \frac{h_o}{2} \right\rfloor \quad (13)$$

Once all the \mathbf{C}' values have been used to restore the original un-watermarked image, the rest of the values in \mathbf{W}' (after removal of the location map \mathbf{L}') belong to the user inserted payload \mathbf{P}' .

3 A novel high capacity reversible watermarking method

A novel reversible watermarking method is described here that improves upon the Tian scheme. The novel method adapts the Tian scheme to incorporate companding of difference values greater than or equal to a user defined threshold.

3.1 Companding technique

In [8], Weng *et al.* presented a reversible watermarking scheme. This scheme used the concept of companding. Companding is the process of **compressing** a signal followed by **expanding** (decompressing³) the signal. Let C be a compression function and let D be a decompressing function. For a signal x , C and D have the following relationship: $D(C(x)) = x$. For a digital signal, C_q and D_q represent the quantized versions of C and D , respectively, and q denotes the quantization function. The compression function C_q is given by:

$$x_q = C_q(x) = \begin{cases} x & |x| < th \\ \text{sgn}(x) \times \left(\left\lfloor \frac{|x| - th}{2} \right\rfloor + th \right) & |x| \geq th \end{cases} \quad (14)$$

³since the term “expanding” is used in this paper to describe the expansion of difference values (h), the **expanding** process of companding will be referred to as decompressing

where $\text{sgn}(\cdot)$ is the sign function and th is a preset threshold chosen by the user. The decompression function D_q is given by:

$$D_q = \begin{cases} x & |x| < th \\ \text{sgn}(x) \times (2|x| - th) & |x| \geq th \end{cases} \quad (15)$$

Companding values via equations 14 and 15 leads to no errors when $|x| < th$. However, when $|x| \geq th$, companding errors r occur:

$$r = |x| - |D_q(C_q(x))| \quad (16)$$

where $r \in \{0, 1\}$.

3.2 Novel method using the companding technique

The novel method presented in this paper treats unaltered and changeable pixel pairs the same way as the Tian method. The novel method differs from the Tian method in the way that it treats expandable pairs; the novel scheme incorporates companding whereas the Tian scheme does not.

The Tian scheme finds expandable pairs using equation 10. As described in Section 2.2.2, if the difference value h of an expandable pair is greater than or equal to a user defined threshold, the pair would be **changed** rather than **expanded**. This was done to minimize the MSE of a watermarked image. As mentioned in Section 2.2.1, changeable pairs do not provide extra storage space; they carry one bit of watermark information in their LSBs but require their original LSB values to be collected in **C** (which constitutes part of the watermark bitstream **W**, as shown in Figure 1).

In the novel scheme, if the difference value h of an expandable pair is greater than or equal to a user defined threshold, the h value is compressed⁴ (via equation 14) and then expanded⁵ (via equation 11). Thus, h values $\geq th$ that would have been **changed** in the Tian scheme are now **compressed** and then **expanded** in the novel scheme. To be able to recover the original image exactly, it is necessary to collect the companding errors (equation 16) into a vector **R**. This vector is embedded into the image as part of the watermark $\mathbf{W} = \mathbf{L} \cup \mathbf{C} \cup \mathbf{R} \cup \mathbf{P}$ (shown in Figure 2). It is important to note that no additional bits of storage are obtained from compressing and expanding as the companding errors have to be stored in **R**. However, having more pixel pairs expanded rather than changed makes the location map more sparse and thus more compressible. This is elaborated upon in Section 3.3.1.

The following MATLAB [12] code snippet shows how the difference value of an expandable pixel pair is

⁴compressed, the first part of companding

⁵values left shifted and watermark bit stored in LSB

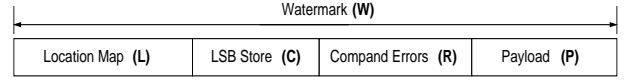


Figure 2. Watermark structure of the novel scheme

companded (using equations 14 and 15) and how the companding error is collected in **R**.

```
% blockIn is a 1x2 non-overlapping block
x=blockIn(1);
y=blockIn(2);
% Find pixel pair average
l = floor( (x+y)/2 );
% Find pixel pair difference value
h = x - y;
% If the difference value is greater
% than or equal to threshold ...
if abs(h) >= th
    % Compress h (companding process)
    hCompress = sign(h) * (floor((abs(h)-th)/2)+th);
    % Use compressed difference value (in later stages)
    h = hCompress;
    % Decompress h (companding process)
    hDecompress = sign(hCompress) * ((2*abs(hCompress))-th);
    % Calculate companding error
    r = abs(hDecompress - h);
    % Store companding error in bitstream R
    R_bitstream=[R_bitstream r];
end
```

3.3 Watermark recovery and restoring the original image exactly

The process used by Tian (described in Section 2.4) to recover the watermark bitstream **W'** from the LSBs of difference values in the watermarked image, is also used in the novel scheme. Once **W'** has been obtained, the location map **L'** can be recovered by identifying the end of message symbol.

The bits in **W'** after the location map **L'** are the stored LSBs of changeable pairs **C'** and the companding errors **R'**. Using equations 1 and 2, the integer transform is applied to the watermarked image I' to obtain l' and h' . Using the stored LSBs (c_i) from **C'**, the companding errors (r_j) from **R** and the location map values, the original difference values h_o can be computed. For each pixel pair:

- If pixel pair are non changeable, do nothing (**unaltered pair**)
- If the pixel pair are changeable and location map is 1 (**expandable pair**)
 - $h_o = \lfloor h'/2 \rfloor$
 - if $(|h_o| \geq th)$

$$h_o = \text{sgn}(h_o) \cdot ((2 \cdot \text{abs}(h_o)) - th + r_j);$$

$$j = j + 1;$$

- If the pixel pair are changeable and location map is 0 (**changeable pair**)

- if $(h' == 0)$ or $(h' == 1)$
 $h_o = 1;$
- else if $(h' == -1)$ or $(h' == -2)$
 $h_o = -2;$
- else
 $h_o = \lfloor h'/2 \rfloor \cdot 2 + c_i;$
 $i = i + 1;$

Applying the inverse integer transform, the restored image values (x_r and y_r) for each pixel pair can be calculated to obtain the original un-watermarked image **exactly**:

$$x_r = l' + \left\lfloor \frac{h_o + 1}{2} \right\rfloor \quad (17)$$

$$y_r = l' - \left\lfloor \frac{h_o}{2} \right\rfloor \quad (18)$$

Once all the **C'** and **R'** values have been used to restore the original un-watermarked image, the rest of the values in **W'** (after removal of the location map **L'**) belong to the user inserted payload **P'**.

3.3.1 Improvement obtained

As stated in Section 3.2, using the companding technique in the novel method does not provide extra bits of storage. The companding technique allows pixels pairs that would have been changed in the Tian scheme to instead be expanded in the novel scheme. This is important as it makes the location map of the novel scheme more sparse than the location map of the Tian scheme. Sparse location maps can be compressed more than non-sparse location maps. Thus, the novel scheme achieves higher payload capacities than the Tian scheme as it produces location maps that are more compressible.

An example of this is given in Figure 3, where binary location maps for the original Tian scheme (a) and the novel scheme (b) are shown⁶. Non-overlapping blocks of 1×2 were used to generate the pixels pairs in these location maps, hence the width is half the height. In Figure 3, white pixels are expandable and black pixels are not. It can be seen that Figure 3(b) is much more sparse than Figure 3(a). Thus, when these location maps are losslessly compressed, greater compression will be achieved from Figure 3(b) than from Figure 3(a). The more compression that can be applied to the location map **L**, the more space there is for the user payload **P**.

⁶original input image was grayscale Lena 512×512 , threshold values of 16 were used in both the Tian and novel schemes

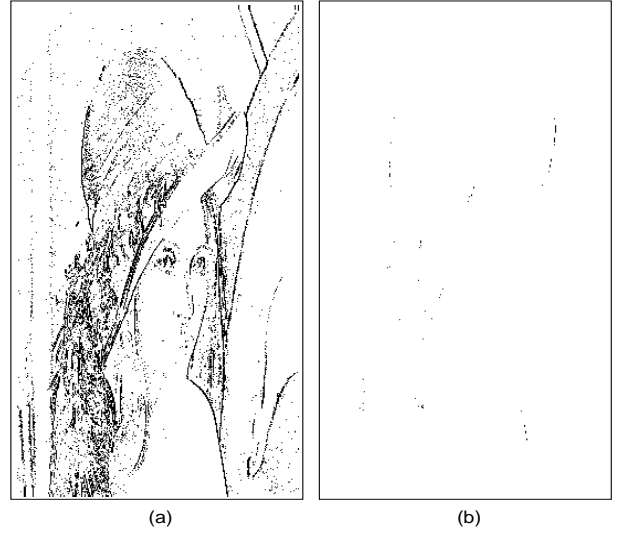


Figure 3. Location maps for (a) Tian scheme (b) new improved scheme; grayscale Lena 512×512 , threshold of 16

The location maps in Figure 3 were stored as binary Portable BitMaps (PBM) and then losslessly compressed via JBIG compression; Table 1 shows the file sizes for each. Clearly, the sparse location map of the novel scheme can be compressed much more than the location map of the Tian scheme.

Scheme	Original PBM	JBIG compressed
Tian method	16396 Bytes	3785 Bytes
novel method	16396 Bytes	101 Bytes

Table 1. Original and compressed file sizes of location maps; grayscale Lena 512×512 , threshold of 16

In summary, the novel scheme (using the companding technique) saves space by producing highly compressible location maps which in turn allows larger user payloads to be inserted.

4 Results

The performance of the novel scheme is compared to that of the Tian scheme using different images. The images used are: (i) Lena (test image), (ii) Eye (SLO⁷ medical image) and (iii) Aerial (SAR⁸ military image).

Initially, for both the novel and Tian schemes, vertical pixel pairs are selected (using non-overlapping blocks of 2×1). In order to achieve payload bitrates greater than 0.5 bits per pixel (bpp), it is necessary to perform multiple embedding. This consists of embedding a watermark

⁷Scanning Laser Ophthalmoscope

⁸Synthetic Aperture Radar

(into image I) using vertical pixel pairs to obtain a watermarked image W_v . Taking advantage of the lossless nature of reversible watermarking, another watermark is embedded into the horizontal pairs of W_v (using non-overlapping blocks of 1×2) to obtain W_{vh} . Thus, the original image I has been watermarked twice resulting in W_{vh} . The original image can be recovered exactly by first decoding W_{vh} to get W_v and then decoding W_v to get I .

Tables 2, 3 and 4 show the results for the Lena, Eye and SAR images, respectively. Each table compares the performance of the Tian and novel schemes via hiding capacity (measured using payload bitrate) and image degradation (measured using Peak Signal to Noise Ratio (PSNR)). Grayscale Lena 512×512 was chosen as the test image as this was the image tested in [3]. Thus, a fair quantitative comparison between the results published in [3] and the results presented here can be obtained.

In Table 2, it can be seen that the novel scheme is performing better than the Tian scheme. For approximately equal payload bitrates, the novel scheme returns better quality (higher PSNR) watermarked images. For example, a payload bitrate of 0.24bpp⁹ in the Tian scheme results in a watermarked image of 42.86dB whereas the same payload bitrate in the novel scheme results in a PSNR of 45.42dB. In general, the novel scheme outperforms the Tian scheme at every bitrate tested. The increase in performance is greater at low bitrates than it is at high bitrates.

The results in Table 3 (for the medical image) follow a similar pattern to those in Table 2, with the novel scheme performing better than the Tian scheme over the complete range of payload bitrates (especially at lower bitrates). For a given bitrate, the PSNR of the novel scheme is higher (better) than the PSNR of the Tian scheme. It can be seen in Table 4 (SAR image) that the novel scheme is again outperforming the Tian scheme over the whole range of payload bitrates. However, the results for both the novel and Tian schemes using the SAR image are not as impressive as the results obtained using the Lena or Eye images. This may be attributed to the fact that both schemes perform better in homogeneous images than they do in inhomogeneous images¹⁰ (in general, homogeneous images will have more expandable pairs).

Figure 4 shows the original Lena, Eye and SAR images. Examples of the bitrates achieved and the corresponding PSNRs using the novel and Tian schemes are given. It can be seen that for approximately equal measures of PSNR, the novel scheme returns much higher bitrates than the Tian scheme.

⁹payload capacity = image size \times bpp = $512 \times 512 \times 0.24 \approx 62914$ bits; the higher the bpp value, the greater the hiding capacity

¹⁰the SAR image is inhomogeneous due to speckle noise

5 Conclusions

A novel reversible watermarking scheme implementing the companding technique has been presented. This novel scheme adapted the Tian scheme to implement companding of certain pixel pairs. This resulted in location maps that were sparser in the novel scheme than they were in the Tian scheme. This in turn allowed the novel scheme to have a higher payload hiding capacity than the Tian scheme (for similar PSNR values). This was verified via experimental results using test, medical and military images.

References

- [1] A. M. Alattar. Reversible watermark using the difference expansion of a generalized integer transform. *IEEE Trans. on Image Processing*, 13(8):1147–1156, 2004.
- [2] J. Fridrich, M. Goljan, and R. Du. Lossless data embedding - a new paradigm in digital watermarking. *EURASIP Journal on Applied Signal Processing*, 2002(2):185–196, Feb. 2002.
- [3] J. Tian. Reversible data embedding using difference expansion. *IEEE Trans. on Circuits and Systems for Video Technology*, 13(8):890–896, August 2003.
- [4] M. U. Celik, G. Sharman, A. M. Tekalp, and E. Saber. Lossless generalised-LSB data embedding. *IEEE Trans. on Image Processing*, 14(2):253–266, 2005.
- [5] C. C. Chang, W. L. Tai, and M. H. Lin. A reversible data hiding scheme with modified side match vector quantization. In *Proc. Intl. Conf. on Advanced Information Networking and Applications*, pages 947–952, Taiwan, March 2005.
- [6] C. D. Vleeschouwer, J. F. Delaigle, and B. Macq. Circular interpretation of bijective transformations in lossless watermarking for media asset management. *IEEE Trans. on Multimedia*, 5(1):97–105, March 2003.
- [7] B. Yang, M. Schmucker, X. Niu, C. Busch, and S. Sun. Reversible image watermarking by histogram modification for integer DCT coefficients. In *Proc. of the 6th IEEE Workshop on Multimedia Signal Processing*, pages 143–146, Siena, Italy, Sept. 2004.
- [8] S. Weng, Y. Zhao, J-S. Pan, and R. Ni. A novel reversible watermarking based on an integer transform. In *Proc. IEEE Intl. Conf. on Image Processing, ICIP'08*, pages 241–244, San Antonio, TX, USA, Sept. 2007.
- [9] D. M. Thodi and J. J. Rodríguez. Reversible watermarking by prediction-error expansion. In *Proc. of the 6th IEEE Southwest Symposium on Image Analysis and Interpretation*, volume 3, pages 21–25, Lake Tahoe, USA, March 2004.
- [10] J-B. Feng, I-C. Lin, C-S. Tsai, and Y-P. Chu. Reversible watermarking: Current status and key issues. *Intl. Journal of Network Security*, 2(3):161–171, May 2006.
- [11] W-K. Kim. The JBIG standard. In R. K. Jurgens, editor, *Digital Consumer Electronics Handbook*, pages 911–924. McGraw-Hill, Inc., Hightstown, NJ, USA, 1997.
- [12] MATLAB [online], Available from the World Wide Web (date accessed: March 2008): <http://www.mathworks.com/products/matlab>.

Tian: payload bitrate (bpp)	0.15	0.24	0.32	0.39	0.46	0.54	0.67	0.74	0.85
Tian: PSNR (dB)	44.20	42.86	41.55	40.06	37.66	36.15	34.80	33.05	32.54
novel: payload bitrate (bpp)	0.16	0.24	0.37	0.41	0.48	0.57	0.70	0.78	0.87
novel: PSNR (dB)	47.09	45.42	42.45	41.22	40.35	38.84	36.78	35.31	33.68

Table 2. Results for the Lena test image; grayscale 512×512

Tian: payload bitrate (bpp)	0.23	0.29	0.34	0.47	0.53	0.66	0.79	0.86	0.94
Tian: PSNR (dB)	42.53	41.62	40.94	37.87	35.97	34.89	33.53	32.73	31.65
novel: payload bitrate (bpp)	0.24	0.29	0.36	0.43	0.60	0.66	0.81	0.89	0.94
novel: PSNR (dB)	47.09	45.27	42.93	40.54	39.61	38.48	35.71	33.77	32.06

Table 3. Results for the Eye medical image; grayscale 256×256

Tian: payload bitrate (bpp)	0.18	0.25	0.33	0.47	0.52	0.64	0.76	0.85	0.90
Tian: PSNR (dB)	36.30	35.29	33.94	29.88	29.50	28.54	27.46	26.58	26.02
novel: payload bitrate (bpp)	0.19	0.25	0.33	0.45	0.54	0.66	0.77	0.85	0.90
novel: PSNR (dB)	40.71	38.74	36.32	34.40	32.87	30.81	28.99	27.41	26.32

Table 4. Results for the SAR military image; grayscale 256×256

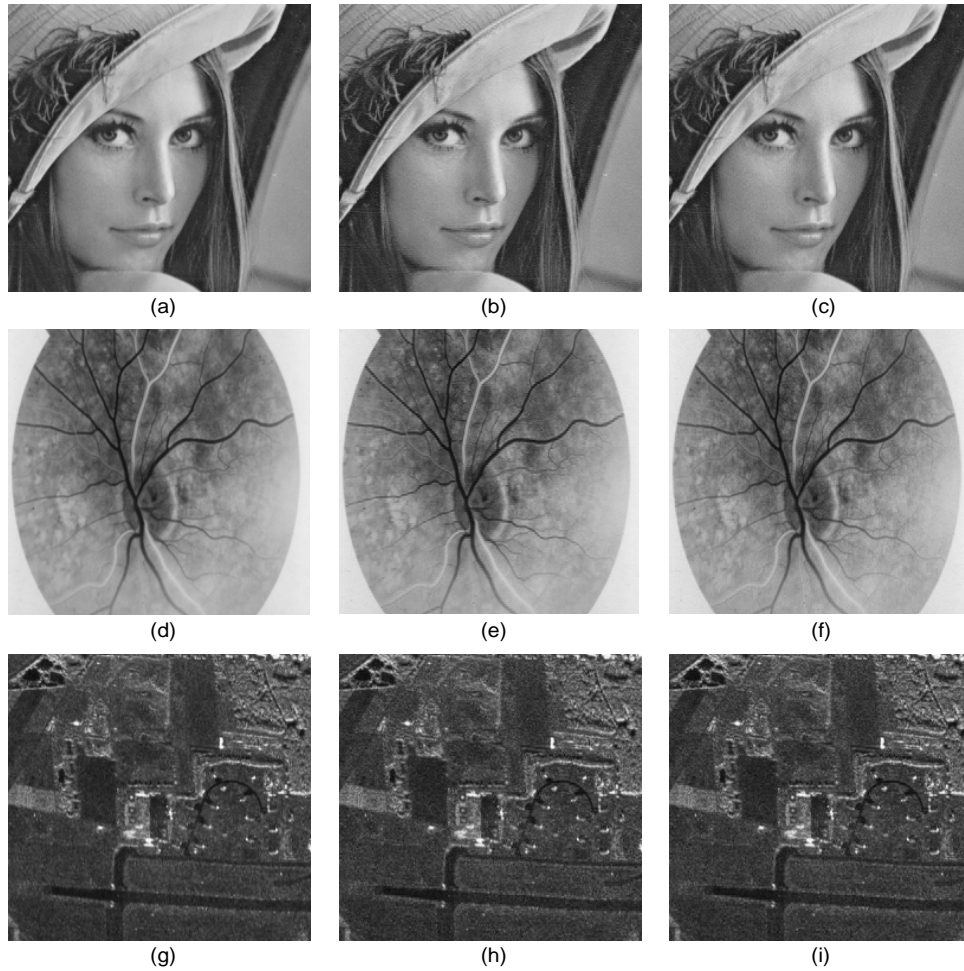


Figure 4. Original images and watermarked images (with approximately equal PSNRs): **(a)** original Lena image (zoomed) **(b)** Tian: payload = 0.39bpp, PSNR = 40.06dB **(c)** novel: payload = 0.48bpp, PSNR = 40.35dB **(d)** original Eye image **(e)** Tian: payload = 0.47bpp, PSNR = 37.87dB **(f)** novel: payload = 0.66bpp, PSNR = 38.48dB **(g)** original SAR image **(h)** Tian: payload = 0.18bpp, PSNR = 36.30dB **(i)** novel: payload = 0.33bpp, PSNR = 36.32dB