

# **An Overview of Transparent and Robust Digital Image Watermarking**

**Guan-Ming Su**

## **I. Introduction**

Due to the rising and flourishing development of network, exchange and distribution of images become much easier and more popular. Hence, the accompanying problem is how to protect the copyright of this information. Ownership and forging-proof are the two key issues in this field. Digital image watermarking is the technique providing embedded copyright information in images and achieving the two goals. In section II of this paper, we discuss the basic requirement that an effective digital watermarking technique should satisfy. Furthermore, we can categorize the fundamental principles of digital watermarking into two categories, spatial and frequency domain. Section III discusses the first technique and Section IV,V and VI survey the frequency technique of DCT, DFT and wavelet domain respectively.

## **II. Requirement of Digital Watermarking**

There are three fundamental goals digital watermarking is supposed to achieve.

1. Transparency: The digital watermark should not degrade the quality of images. In other words, human visual system will not be disturbed by embedded watermark.
2. Robustness: Counterfeiters would try to modify or attack the watermarked images. Therefore, these watermarks should be robust enough to defeat these attacks. Petitcolas [1] catalogs these attacks as follows:
  - JPEG Compression.
  - Geometric transform: horizontal flip, rotation, cropping, scaling, deletion of lines or columns, generalized geometrical transformations, random geometric distortions, geometric distortions with JPEG
  - Enhancement techniques: low pass filtering, sharpening, histogram modification, gamma correction, color quantization, restoration
  - Noise addition
  - Printing-scanning
  - Statistical averaging and collusion
  - Over-marking
  - Oracle attack
3. Capacity: These embedded watermarks should carry enough information to represent their unique identities.

## **III. Spatial Domain Digital Watermarking**

Most of early researches of digital watermarking technique embedded the watermark in the spatial domain. They modified the least significant bit (LSB) by some techniques, such as checksum, adding identification string. Basically, these kinds of watermarks are very fragile. They could be changed by noise, compression and interpolation. Therefore, some refined methods were proposed.

### **A. 1-D M-Sequence Approach**

M-sequence has noise-like characteristic, resistance to interference, and good auto-correlation properties (i.e. a single peak without side-lobes), and can be easily generated by a linear shift register. Suppose we have an image  $X$  with  $L*L$  pixels, [2] suggests an encoding procedure that we can generate a 1-D m-sequence  $m$  with length  $L$  according to owner's authorized key  $K$ . The watermark  $W$  consists of  $L$  shifted copies of  $m$ . The watermarked image  $Y$  can be expressed as  $Y=X+W$ . Let  $Y'$  denote the published image. The decoding procedure uses the key to obtain  $m$  and calculates the auto-correlation between  $m$  and each row  $y$  of  $Y'$ . If there is few or no peak, we can claim  $Y'$  is a forged version. There are some refined versions that use extended m-sequence, such as Gold or Kasami codes.

### **B. 2-D M-Sequence Approach**

In [3], the authors propose another 2-D version approach. First, we generate a 1-D m-sequence and divide it into segments with length 64. Resize each segment to  $8*8$  and tile them to host image. The verification process uses the same technique in 1-D case to compute the auto-correlation. These two methods are resilient to noise.

### **C. Salient-Point Modification**

Instead of globally embedding watermark, [4] proposes that we could embed these watermarks in the salient points of an image  $I$ . Salient points are defined as pixels with highest values for a given function  $F$ . Consequently, these points could be corners or locations with high entropy. After choosing the salient pixels  $S$ , we use a key  $K$  to generate dense pseudo-random subset of pixels  $W$  which  $C= W \cap S$  and  $|C|/|S| > 0.5$  for watermarking. In addition, we embed digital watermark in the pixels of  $C$ . The detection method first uses the function  $F$  and key  $K$  to generate  $S'$  and  $W$ . Let  $S_1, S_2$  denote  $W \cap S$  and  $(I-W) \cap S$  respectively. Next, check whether  $|S_1|-|S_2|$  is greater than a given threshold or not to determine the authorization.

In general, spatial domain watermarking technique is not robust. If the attackers know these watermarks are embedded in LSB, they can randomly rearrange structures of these LSB and lose the watermarks. Although some methods, e.g. in [5], adopted pre-filtering skill to increase the percentage of identification, experimental results have shown the fundamental disadvantages of spatial domain watermarking.

## **IV. DCT Domain Digital Watermarking**

Several techniques can transform an image into frequency domain, such as DCT, DFT and wavelet. Each transform has its advantages and disadvantages. First of all, we discuss the DCT approach.

### **A. Global DCT Domain Watermarking**

### (i). Coherent Decoding Method

I.J. Cox et al. in [6] suggest another point of view that the watermarks should be embedded in the significant perceptual component of HVS. The major reason is most compression techniques try to reduce redundancy of images. In other words, they modify the insubstantial part, such as LSB in spatial domain and high frequency in frequency domain. This principle can explain why these early works are not robust. Moreover, [6] supposes the original non-watermarked image can be obtained in the decoding part. Therefore, they propose a spread spectrum watermarking technique as follows:

Encoding procedure:

1. Given an  $N \times N$  image  $D$ , we implement  $N \times N$  DCT and obtain  $N \times N$  coefficients.
2. Pick the largest  $n$  coefficients as a vector  $v$ .
3. Generate a vector  $x$  with length  $n$  by Gaussian  $N(0,1)$ . Every key owns a unique  $x$ .
4. Select a scaling factor  $\alpha$  and construct a new vector  $v'$  with

$$v'_i = v_i(1 + \alpha x_i)$$

5. Replace the original  $v$  by  $v'$  and perform IDCT to obtain a watermarked image  $D'$ .

The decoding procedure:

1. Perform DCT with  $D'$  and obtain  $v^*$  whose elements correspond to the coordinates of  $v$  in  $D$ .
2. Calculate  $x^* = v^* - v$ .
3. Measure the similarity between  $x^*$  and original key  $x$

$$\text{sim}(x, x^*) = \frac{(x^*)^T x}{\sqrt{(x^*)^T x^*}}$$

4. Verify whether the similarity is greater than a preset threshold and determine the copyright.

[6] has shown the strong robustness of this technique, even multiple-document attacks. As a matter of fact, no previous works can achieve its ability. However, the primary disadvantage is the need to synchronize with the unmarked image in the decoder. It's not practical that a client wants to confirm copyright of an image but he cannot obtain the original image.

### (ii) Non-coherent Decoder Method

M. Barni et al. in [7] modify some details of the previous work under the same framework. This paper indicates that if we only select the largest  $n$  coefficients as a vector  $v''$  from the watermarked image  $D'$  without the help of original image  $D$ , it's difficult to guarantee  $v^* = v''$ . As a result, they propose that we can zigzag the order of DCT coefficients (like JPEG) and pick the  $(L+1)^{\text{th}}$  to  $(L+M)^{\text{th}}$  for watermarking. To decrease the error probability, they suggest the second term in watermarked vector should be proportional to the absolute value of coefficients.

$$v_{L+i} = v_{L+i} + \mathbf{a} | v_{L+i} | x_{L+i}$$

The watermark detection can be first done by extracting a vector  $v^{**}$  from the  $(L+1)^{\text{th}}$  to  $(L+M)^{\text{th}}$  coefficients of DCT  $D'$ . Calculate the similarity between  $v^{**}$  and the key  $x$  by the following formula and compare it with a given threshold.

$$z = \frac{x^T(v^{**})}{M}$$

Experimental results also show its robustness.

## B. Block DCT Domain Watermarking

Because of the robustness in [6], later researchers present some modified techniques. To increase the capacity of information and detection for cropping and localized signal processing, some works suggest blocking the images and performing DCT, which is the same way as JPEG.

### (i) Identical Masking of DCT Block

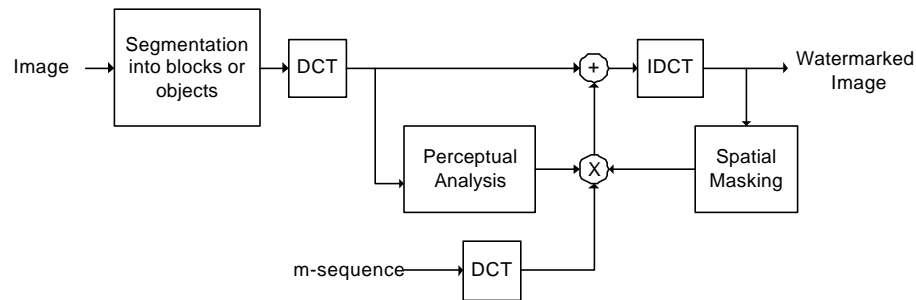


Figure 1

The above figure shows the structure of the encoder proposed by [8]. The watermark generation procedure is as follows:

1. Block the image into  $B_i$ , perform DCT of each block and obtain  $B_i'$ .
2. Calculate the frequency mask  $FM$  by a visual model.
3. Generate different m-sequences for each block and perform DCT to obtain  $W_i$ .
4. Masked coefficients in each  $B_i'$  are multiplied by  $W_i$ , i.e.  $M_i = W_i * (FM * B_i')$ .
5. Perform inverse DCT of  $M_i$  and scale it by spatial mask to control the invisibility of image.

The reason we adopt different m-sequences in this approach is that we can reduce unauthorized attack by removing cross-correlation of each block. Frequency and spatial masks assure the transparency of image. However, this method needs the original image in decoder to detect the watermark.

### (ii) Image Adaptive Masking of DCT Block

In fact, the scheme in figure 1 is equal to frequency weighting in each DCT block. However, each block contains the same amount of information. To increase the capacity, in [9] the watermark can be adaptively embedded in each block according to local image characteristics. The measurement of characteristics is obtained from *just noticeable differences* (JND) of Waton's Model [10].

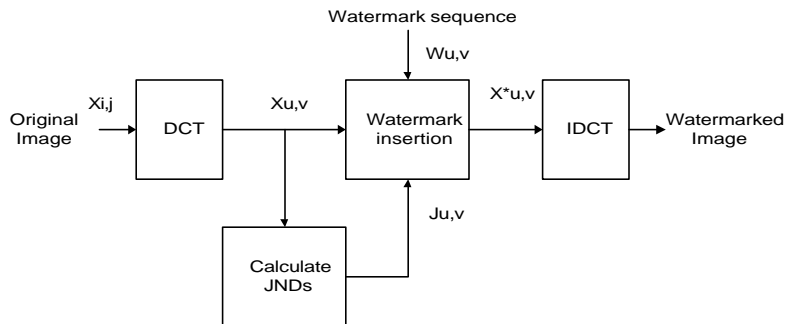


Figure 2

The block diagram is shown in Figure 2. First, perform DCT of each block of original image and obtain coefficients  $X_{u,v,b}$ .  $(u,v)$  is the coordinates of  $b^{\text{th}}$  DCT block. Second, generate an independent Gaussian sequence,  $w_{u,v,b}$ , with  $N(0,1)$  according to a given user key. The embedding procedure is:

$$X_{u,v,b}^* = \begin{cases} X_{u,v,b} + t_{u,v,b} w_{u,v,b} & \text{if } X_{u,v,b} > t_{u,v,b} \\ X_{u,v,b} & \text{otherwise} \end{cases}$$

where  $t_{u,v,b}$  is the JND threshold calculated from Waton's Model. And the detection procedure:

$$W_{u,v,b}^* = \frac{X_{u,v,b} - \hat{X}_{u,v,b}^*}{t_{u,v,b}} \quad d_{WW^*} = \frac{w^* \cdot w}{\sqrt{E_{w^*} E_w}}$$

where  $E_w = w \cdot w$  and  $E_{w^*} = w^* \cdot w^*$ .

After calculating the normalized correlation,  $d_{ww^*}$ , between original and watermarked images and comparing it with a threshold, we can determine whether the watermark is detected or not. Based on the framework, the authors also use wavelet filter to replace DCT. They obtain a much better improvement comparing with I.J. Cox in [6]. However, the common disadvantage of DCT domain techniques is the lack of resilient from geometric transform attacks.

## V. DFT Domain Digital Watermarking

Fourier transform has some integral transform-based invariants properties. For instance, shifting in the spatial domain causes linear shifting in frequency domain, scaling the axes in the spatial domain causes an inverse scaling in the frequency domain, and rotating an image by an angle in the spatial domain causes the

same angle in frequency domain. These properties could help watermark to resist geometric transform attacks.

### A. Log-Polar Mapping Method

[11] proposes that we could adopt Fourier-Mellin Transform and log-polar mapping (LPM), i.e. given a point(x,y) of image, define

$$x = e^m \cos q \quad y = e^m \sin q$$

The transformed coordinate system has some useful properties:

$$(rx, ry) \leftrightarrow (m + \log r, q)$$

$$(x \cos(d) - y \sin(d), x \sin(d) + y \cos(d)) \leftrightarrow (mq + d)$$

The Fourier-Mellin transform is:

$$F_M(k_1, k_2) = \int_{-x}^x \int_0^{2p} f(e^m \cos q, e^m \sin q) \exp(j(k_1 m + k_2 q)) dm dq$$

[11] shows these properties in theory can recover the watermark even it is rotated and scaled. Based on these fundamental properties, they propose a framework to embed watermarks as shown in Figure 3.

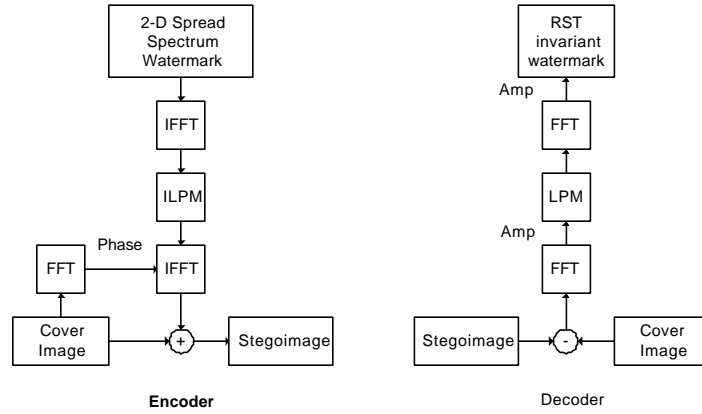


Figure 3

They suggest that log-polar mapping needs interpolation after changing coordinate system and numerical problem will result in degrading image quality. Therefore, they only perform these properties on watermark itself.

However, these are several disadvantages in this scheme. First, this approach requires original image in the decoder. Second, it cannot resist aspect ratio and shear transformations attack. In addition, it has less ability to withstand JPEG compression and cropping attack.

## B. Template Based Method with Log-Polar and Log-log Mapping

The authors in [12] try to combine the good detection of rotation and scale of log-polar mapping and the scheme in [6]. They embed watermark in the DFT mid-band domain and a template in the low frequency part. Under this framework, the decoding procedure does not require the original image. It can detect the template by log-polar mapping and recover the coordinate system to extract the watermark. However, this framework cannot deal with aspect ratio attack. They propose another mapping, log-log mapping, to cope with this attack.

$$(x, y) \in R^2 \quad x = e^{m_1} \quad y = e^{m_2} \quad \mathbf{m}_1, \mathbf{m}_2 \in R$$

Unfortunately, log-log mapping cannot resist rotation and scale attack.

## C. Template Matching Method

Instead of embedding template in the log-polar or log-log coordinate systems, [13] proposes a simpler but more robust method. Both the watermark (BCH codes) and template are embedded in the DFT domain. The template consists of two lines (each line consists of 7 pixels) distributed from radius  $f_1$  to  $f_2$  and angle  $\theta_1$  to  $\theta_2$ . The radii and angles are generated by user key. Pixels on the lines have the values that are equal to the local average value of DFT points plus two standard deviations. The decoder can detect these two lines and extract watermarks by inverse rotation or scaling. Although this technique can perfectly extract watermark by rotation and shear attack (detection probability is 1), its compression and scaling performance are not good (detection probability is about 0.7~0.8).

## D. Circularly Symmetric Watermark

Instead of embedding templates, [14] suggests we can embed a circularly symmetric watermark  $W(k_1, k_2)$  in DFT domain. Let  $I(k_1, k_2)$ ,  $M(k_1, k_2)$  denote the DFT of image and DFT magnitude respectively. The watermark consists of  $p$  rings with radii  $R_i$  and  $R_{i+1}$ ,  $i=1, 2, \dots, p$ . Each ring is distributed as follows:

$$W(r, \mathbf{q}) = \begin{cases} 0 & \text{if } r < R_i \text{ and } r > R_{i+1} \\ \pm 1 & \text{if } R_i < r < R_{i+1} \end{cases}$$

$$r = \sqrt{k_1^2 + k_2^2} \quad \mathbf{q} = \arctan\left(\frac{k_2}{k_1}\right)$$

Furthermore, each ring is divided into  $s$  sectors. As a result, there are  $ps$  sectors in this watermark.

However, to maintain the real value of image after inverse DFT, the watermark must be symmetric.

Therefore, there are  $ps/2$  sectors in the watermark (i.e.  $W(k_1, k_2) = W(N-k_1, N-k_2)$ ). The watermarked image is simply to combine  $M(k_1, k_2)$  with  $W(k_1, k_2)$ , i.e.  $M'(k_1, k_2) = M(k_1, k_2) + W(k_1, k_2)$ , and take inverse DFT of  $M'(k_1, k_2)$ .

The detection procedure simply calculates the correlation,  $c$ , between  $M'(k_1, k_2)$  and  $W(k_1, k_2)$ , compares it with a given threshold, and determines whether it's a watermarked image or not.

$$c = \sum_{k_1=1}^N \sum_{k_2=1}^N W(k_1, k_2) M'(k_1, k_2)$$

Since watermark is circularly symmetric, it is resilient to geometric attacks, such as rotation. In addition, its DFT properties help resist translation and scaling attacks.

## VI. Wavelet Domain Digital Watermarking

Wavelet plays a more and more important role in contemporary image processing field. It has lots of special advantages that conventional transforms, such as DCT and DFT, cannot achieve. Furthermore, it has become the fundamental transform in JPEG2000 standard.

### A. Image Adaptive Masking of DCT Block

Authors in [9] also propose a wavelet version under the same framework. They hierarchically decompose an image into 4 layers by 9-7 orthogonal filters. The watermark is embedded as follows:

$$X_{u,v,l,f}^* = \begin{cases} X_{u,v,l,f} + t_{u,v,l,f} w_{u,v,l,f} & \text{if } X_{u,v,l,f} > t_{u,v,l,f} \\ X_{u,v,l,f} & \text{otherwise} \end{cases} \quad \text{for } l = 1, 2, 3, 4; \quad f = 1, 2, 3$$

where  $t_{u,v,l,f}$  is the JND threshold calculated from Waton's Model. And the detection procedure:

$$w_{u,v,l,f}^* = \frac{X_{u,v,l,f} - \hat{X}_{u,v,l,f}^*}{t_{u,v,l,f}} \quad \mathbf{r}_{ww^*}(l, f) = \frac{w_{l,f}^* \cdot w_{l,f}}{\sqrt{E_{w_{l,f}^*} \cdot E_{w_{l,f}}}}$$

$$\mathbf{r}_{ww^*}(l) = \frac{1}{N_f} \sum_{f=1}^{N_f} \mathbf{r}_{ww^*}(l, f) \quad \text{for } l = 1, 2, 3, 4$$

$$\mathbf{r}_{ww^*}(f) = \frac{1}{N_l} \sum_{l=1}^{N_l} \mathbf{r}_{ww^*}(l, f) \quad \text{for } f = 1, 2, 3$$

$$\mathbf{r}_{ww^*}^* = \max_{l,f} \{ \mathbf{r}_{ww^*}(l), \mathbf{r}_{ww^*}(f) \}$$

where  $X_{u,v,l,f}$  is the coefficient with position  $(u, v)$  in resolution level  $l$  and frequency orientation  $f$ .  $X_{u,v,l,f}^*$  is the corresponding watermarked coefficients.  $w_{u,v,l,f}$  is the watermark sequence. By comparing  $\mathbf{r}_{ww^*}^*$  with a given threshold, we can determine whether a watermark is detected or not.

As shown in the last three formulas, the authors calculate normalized correlations for each frequency and layer and select the maximal value from these correlations. The basic idea is to combine advantages of spatial properties with frequency ones. If a forger wants to crop the image, the higher-level detection suffers less attack than the lower-level one. On the other hand, if a forger wants to smooth the image, lower level detection can resist such an attack.

This approach performs much better than its DCT version, for example, scaling, even cropping with compression. However, it needs the original image and suffers the same fragility by rotation, the common drawback for DCT version.

## B. Self-Similar Circularly Symmetric Watermarking

[15] suggests that we can combine both advantages of wavelet and circularly symmetric watermark. First, we generate a mother circularly symmetric pattern with the structure in [14] and then shift and scale the mother function to obtain a family of patterns.

$$W(n_1, n_2) = \sum_{f=0}^3 \sum_{i=0}^{2^f-1} \sum_{j=0}^{2^f-1} W_M(2^f \cdot n_1 + i \cdot \text{ceil}(\frac{r_{\max}}{2^f}), 2^f \cdot n_2 + j \cdot \text{ceil}(\frac{r_{\max}}{2^f}))$$

The image is decomposed into 4-level by Haar wavelet transform. The watermark is embedded in the 1<sup>st</sup> level and a scaling 1/2 version of the same watermark is embedded in the 2<sup>nd</sup> level. The watermarked image is obtained after inverse wavelet transform. Figure 4 shows the whole framework. The detection procedure calculates the correlation between watermarked coefficients and watermark itself.

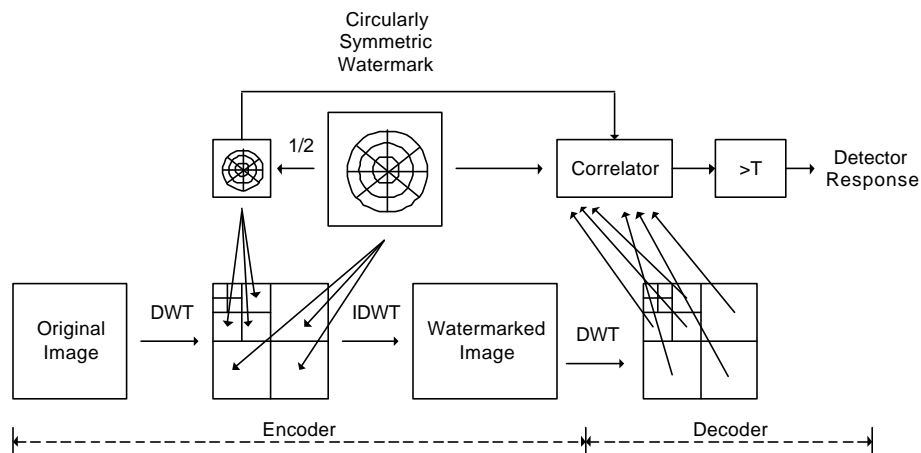


Figure 4

## VII. Conclusion

Intelligence property and copyright protection play important roles in contemporary society. They contribute the progress of technologies. As we have seen in previous sections, modern researchers of digital watermarking prefer frequency domain technique. However, no current technique satisfies the requirement stated in Section II. It doesn't mean that it is impossible to design a robust digital watermarking. In contrast, it's a burgeoning area and it still needs more researchers contribute themselves to explore more reliable techniques.

## VIII. Reference:

- [1] M. Kutter and F.A.P. Petitcolas, "A fair benchmark for image watermarking systems," in *Proc. Electronic Imaging '99, Security and watermarking of Multimedia Contents*, vol 3657, San Jose, CA, Jan. 25-27, 1999, pp226-239.
- [2] R.G. van Schyndel, A.Z. Tirkel, N. R. A. Mee, C.F. Osborne, "A digital watermark," in *Proceedings of the International Conference on Image Processing*, November, 1994, Austin, Texas, vol. 2, pp. 86-90.
- [3] R. B. Wolfgang and E. J. Delp, "A Watermarking Technique for Digital Imagery: Further Studies," in *Proceedings of the International Conference on Imaging Science, Systems, and Technology* June 30-July 3, 1997, Las Vegas, pp. 279-287.
- [4] P.M.J. Rongen, M.J.J.B. Maes, and C.W.A.M. van Overveld, "Digital image watermarking by salient point modification," in *Proc. SPIE Electronic Imaging '99, Security and watermarking of Multimedia Contents*, vol 3657, San Jose, CA, Jan. 25-27, 1999, pp273-282.
- [5] G. Depovere, T. Kalker and J.-P. Linnartz, "Improved watermark detection reliability using filtering before correlation," in *Proc. 5<sup>th</sup> IEEE Int. Conf. Image Processing, ICIP 98*. vol 1, Chicago, IL Oct 4-7, 1998, pp: 430-434
- [6] I.J Cox, J. Kilian, F.T. Leighton, and T. Shamoan, "Secure spread spectrum watermarking for multimedia" in *IEEE Transactions on Image Processing*, vol: 6 Issue: 12, Dec.1997, pp:1673 -1687
- [7] Mauro Barni, Franco Bartolini, Vito Cappellini and Alessandro Piva, "A DCT-domain system for robust image watermarking," in *Signal Processing*, vol 66 May 1998, pp.357-372
- [8] M.D. Swanson, Bin Zhu, A.H.Tewfik, "Transparent robust image watermarking," in *Proceedings of International Conference on Image Processing*, vol: 3, 1996, pp 211 -214
- [9] C.I Podilchuk and Wenjun Zeng, "Image-adaptive watermarking using visual models," in *IEEE Journal on Selected Areas in Communications*, vol: 16 Issue: 4 , May 1998, pp: 525 -539
- [10] A. B. Watson, "DCT quantization matrices visually optimized for individual images," in *Proc. SPIE Conf. Human Vision, Visual Processing and Digital Display IV*, Feb. 1993, vol. 1913, pp. 202-216.
- [11] J.J.K.Ó Ruanaidh and T. Pun, "Rotation, scale and translation invariant spread spectrum digital image watermarking," in *Signal Processing*, vol 66 May 1998, pp. 303-317
- [12] S. Pereira, J.J.K.Ó Ruanaidh, F. Deguillaume, G.Csurka and T. Pun, "Template based recovery of Fourier-based watermarks using log-polar and log-log maps," in *1999. IEEE International Conference on Multimedia Computing and Systems*, vol: 1 , 1999 pp: 870 -874
- [13] Shelby Pereira and Thierry Pun, "Robust Template Matching for Affine Resistant Image Watermarks," in *IEEE Transactions on Image Processing*, vol 9, no 6, June 2000, pp1123-1129
- [14] V. Solachidis and I. Pitas, "Circularly symmetric watermark embedding in 2-D DFT domain," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing 1999*, vol: 6 , pp: 3469 -3472
- [15] S. Tsekeridou and I. Pitas, "Wavelet-based self-similar watermarking for still images," in *Proceedings of IEEE International Symposium on Circuits and Systems, 2000*, vol: 1 pp: 220 -223