A Watermarking Approach Using Balanced Incomplete Block Designs for Watermark Design

Hsiu-Te Su　　　Zhi-Fang Yang

Department of Computer Science and Information Engineering, National Taipei University
151, University Rd., San Shia, Taipei, 237 Taiwan, R. O. C.
zfyang@mail.ntpu.edu.tw

Abstract

Digital copyright protection has become a more and more serious problem. Digital watermarking is a potential solution by which copyright information can be embedded into digital contents. This work aims to design watermarks based on the balanced incomplete block designs (BIBD). The mathematical structure of BIBD is used to construct the copyright mark and to facilitate robustness of the embedded watermark. Experimental results show the feasibility of the proposed approach.

Keyword: digital copyright protection, balanced incomplete block designs (BIBD).

1 Introduction

Due to the convenience of duplication of multimedia data, digital copyright protection has become a more and more serious problem. Digital watermarking is a potential solution since copyright information can be imperceptibly embedded within the host data [1-10, 12-21].

Many researchers have attempted to design generic models of watermarking [9]. Regardless of which models are adopted, watermark design is the first essential step [1, 4]. In previous works, orthogonal patterns or random sequences are generally preferred for good code separation [1]. However, massive storage and long codes are usually required for practical problems. Furthermore, computation of correlations grows along with the size of the message set. Besides, the conflict is worse between the requirement of robustness and the need of imperceptibility.

In this work, to tackle the above problems, messages are designed based on a mathematical structure called balanced incomplete block designs (BIBD) [11]. Mathematical properties of BIBD are utilized to generate a set of basic symbols. A set of coding symbols is then created by concatenating a certain number of the basic symbols according to the desired number of coding symbols. The copyright information is encoded by a sequence of coding symbols, which is actually a matrix of binary values. The matrix is then scanned into a one-dimensional vector, then the copyright watermark is obtained by randomly permuting the one-dimensional vector. To recognize the transmitted watermark, inverse process of the watermark design stage is performed to get a sequence of extracted basic symbols. The mathematical structure of BIBD is then used to design rules for watermark recognition.

The rest of this paper is organized as follows. Section 2 gives the proposed approach of BIBD-based watermark design and the corresponding detection scheme. Section 3 offers experimental results on a watermarking approach based on the proposed BIBD-based watermarks. Conclusions are finally made in Section 4.

2 BIBD-based Watermarking

2.1 Basics of BIBD

Balanced incomplete block designs (BIBD) is a mathematical structure consists of a pair (V, β), where V is a *v*-set, and β a collection of *k*-subsets of *V* (blocks) [11]. Five parameters are used to define a BIBD, denoted as a (v, b, r, k, λ)-BIBD [11], where the number of elements in V is v, the number of blocks in β is b, the number of elements in each block is exactly k, each element in V is contained in exactly r blocks, and every 2-subset in V is contained in exactly λ blocks [11]. By using general counting principles, the three parameters v, k, and λ can be used to determine the other two parameters as follows [11].

$$r = \frac{\lambda(v-1)}{k-1} \tag{1}$$

and

$$b = \frac{vr}{k}. \tag{2}$$

Therefore, one can usually denote a (v, b, r, k, λ) -BIBD as a (v, k, λ) -BIBD.

The incidence matrix of a BIBD is used in the proposed approach. The incidence matrix is a unique representation of a (v, k, λ)-BIBD, which is a $v \times b$ matrix $A = \left(a_{ij}\right)$ with $a_{ij} = 1$ if the ith element in V occurs in the jth block in β, and $a_{ij} = 0$ otherwise. For example, Eq.(3) is the incidence matrix A of the (6, 3, 2)-BIBD with $k = 3$ and $r = 5$.

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 \end{bmatrix} \tag{3}$$

, where each column contains three 1's, and each row contains five 1's.

2.2 Watermark Design

In the proposed approach, a (v, k, λ)-BIBD is first used to generate a set of basic symbols, which are then used to encode a set of coding symbols. The basic symbols, denoted as c_i, is created by choosing λ blocks in the BIBD set with the chosen λ blocks composed of exactly one same pair of elements from β. When the basic symbols are defined, the coding symbols are created by concatenating the basic symbols. The number of basic symbols used for one coding symbol is determined according to the available basic symbols and the desired number of coding symbols. For example, if four basic symbols are available and sixteen coding symbols are desired, two or more basic symbols would be used to design one coding symbol.

The copyright information, which is represented by a sequence of coding symbols, is encoded into a matrix just by concatenating corresponding coding symbols. The matrix is then scanned into a one-dimensional vector in row-major order. Finally, the copyright watermark is obtained by randomly permuting the one-dimensional vector.

The following is an example of the whole process of watermark design based on a (7, 3, 2)-BIBD. There are four BIBDs with the parameter set (7, 3, 2) [11], one incidence matrix is shown in Eq.(4). Since $\lambda = 2$, two blocks are chosen to create one basic symbol, in which an identical pair of elements contained for both the blocks. Four basic symbols are designed as shown in Eq.(5). A set of sixteen coding symbols is designed by concatenating two basic symbols. The basic idea of choosing basic symbols is to make the hamming distances of the coding symbols as large as possible. Twenty-eight bits are required for each coding symbols. Then the sixteen coding symbols are used to represent the sixteen hexadecimal digits as shown in Eq.(6). Now hexadecimal numbers can be encoded, for example, the number 0x01F can be encoded as shown in Eq.(7), and the corresponding one-dimensional vector is shown in Eq.(8). Finally, the watermark is obtained after performing random permutation by using a secret key recording the way of permutation. The watermark and the permutation key then is embedded and transmitted to the receiver site.

$$(7,3,2) = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix} \tag{4}$$

$$c_0 = \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix},\ c_1 = \begin{bmatrix} 0 & 0 \\ 1 & 1 \\ 0 & 0 \\ 1 & 0 \\ 1 & 1 \\ 0 & 1 \\ 0 & 0 \end{bmatrix},\ c_2 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \\ 1 & 1 \\ 0 & 0 \\ 1 & 1 \\ 0 & 0 \end{bmatrix},\ and\ c_3 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 1 \\ 1 & 1 \end{bmatrix} \tag{5}$$

$$\begin{aligned} 0 &= \begin{bmatrix} c_0 \\ c_0 \end{bmatrix}, 1 = \begin{bmatrix} c_0 \\ c_1 \end{bmatrix}, 2 = \begin{bmatrix} c_0 \\ c_2 \end{bmatrix}, 3 = \begin{bmatrix} c_0 \\ c_3 \end{bmatrix}, \\ 4 &= \begin{bmatrix} c_1 \\ c_0 \end{bmatrix}, 5 = \begin{bmatrix} c_1 \\ c_1 \end{bmatrix}, 6 = \begin{bmatrix} c_1 \\ c_2 \end{bmatrix}, 7 = \begin{bmatrix} c_1 \\ c_3 \end{bmatrix}, \\ 8 &= \begin{bmatrix} c_2 \\ c_0 \end{bmatrix}, 9 = \begin{bmatrix} c_2 \\ c_1 \end{bmatrix}, a = \begin{bmatrix} c_2 \\ c_2 \end{bmatrix}, b = \begin{bmatrix} c_2 \\ c_3 \end{bmatrix}, \\ c &= \begin{bmatrix} c_3 \\ c_0 \end{bmatrix}, d = \begin{bmatrix} c_3 \\ c_1 \end{bmatrix}, e = \begin{bmatrix} c_3 \\ c_2 \end{bmatrix}, f = \begin{bmatrix} c_3 \\ c_3 \end{bmatrix}. \end{aligned} \tag{6}$$

$$\text{0x01F} = \begin{bmatrix} c_0 \\ c_0 \\ c_0 \\ c_1 \\ c_3 \\ c_3 \end{bmatrix} \tag{7}$$

$$\begin{aligned} \text{0x01F} = [&11111001000000011111001000000 \\ &11111001000000000110010110100 \\ &1001000000111110010000001111] \end{aligned} \tag{8}$$

2.3 Watermark Recognition

Once the embedded watermark is extracted at the receiver site, the corresponding inverse permutation is performed. To recognize the transmitted watermark, first re-assemble the one-dimensional binary sequence into a matrix with exactly λ columns. Then divide the matrix into a set of $v \times \lambda$ matrices, and each $v \times \lambda$ matrix represents an extracted basic symbol. Each extracted basic symbol is recognized based on the rules proposed in this study. Finally the decoded hexadecimal digits are constructed to get the number representing the copyright information.

The mathematical structure of BIBD is used to design the three rules of watermark recognition. Every basic symbol is treated as a candidate. Scores are computed for each candidate basic symbols based on three rules. The basic symbol with the highest score is selected as the extracted basic symbol. The three rules are described as follows, respectively.

The first rule is to check whether the pair contained in the candidate basic symbol is in the blocks of the extracted basic symbol. The score is added by one if the pair is found. The score is added by 0.5 if half of the pair is found, that is, only one bit of the pair. Otherwise the score is not changed. Score in this part is given based on the single pair. An example is shown in Fig. 1.

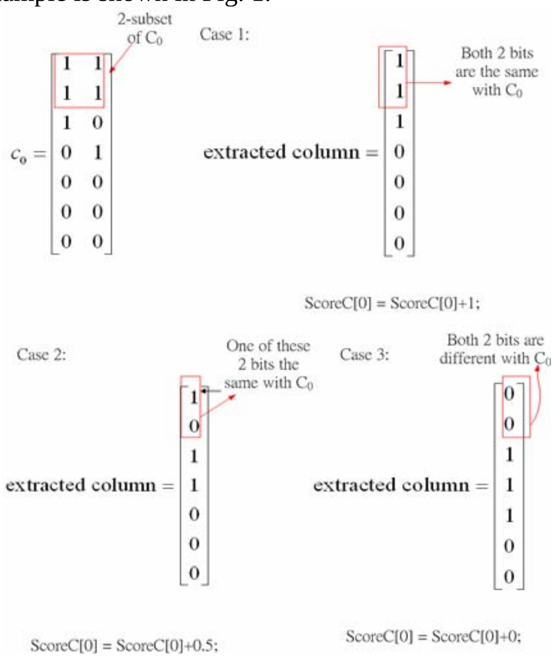

Fig.1 Example of score computation of the first rule.

The second rule tends to check whether the bits of the pair, total ($\lambda \times 2$) bits, in the candidate basic symbol are shown in the extracted basic symbol. The score is added by $1/(\lambda \times 2)$ for each corresponding bit. Score in this part is given based on the whole λ pairs. An example is shown in Fig. 2.

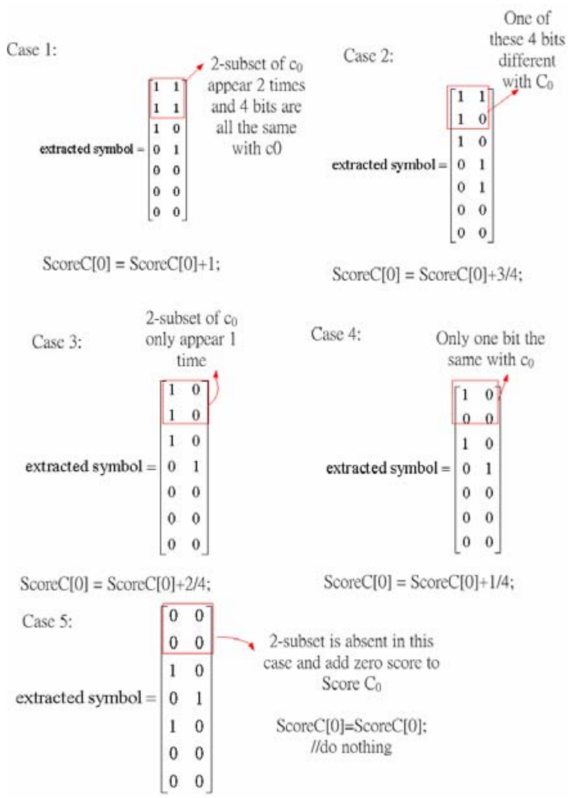

Fig.2 Example of score computation of the second rule.

The goal of the third rule is to check the whole extracted basic symbol based on correlation. The score is added by $1/v$ for each corresponding bit. The highest score obtained in this part for one candidate basic symbol is λ. An example is shown in Fig. 3.

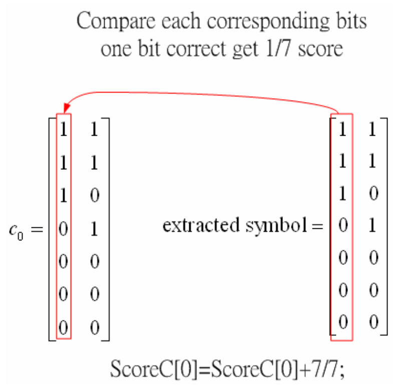

Fig.3 Example of score computation of the third rule.

2.3 Watermark Embedding and Extraction

The watermarking process is divided into DWT decomposition, HVS analysis, and watermark embedding. This part totally follows the approach proposed in [10] except that a four-level Haar transform is used instead. The three detail sub-bands, are used to embed the watermark as in [10]. The main reasons are to increase data payload and better invisibility, and the HVS analysis is used to guarantee robustness as discussed in [10].

The original image is used to extract the embedded watermark. When the watermarked image is received, both the transmitted image and original image go through DWT. Subtraction is then applied to get the difference matrix. The sign of coefficient difference in the difference matrix is used to determine the extracted watermark bit; that is, a negative number in the difference matrix means the corresponding watermark bit is zero, and otherwise the extracted watermark bit is one. Then the embedded watermark is extracted.

3 Experimental Results

The algorithm was tested on various standard images, using various attacks. The BIBD coding was compared with random generation coding and orthogonal coding. The random generation coding was performed to generate four binary sequence patterns randomly, and to ensure that they had the same size as BIBD basic symbols. The four patterns were then encoded using the method described earlier. The orthogonal coding also generated four patterns with the same size of BIBD basic symbols. The four patterns had to be orthogonal with each other. The decoding step of later two codes was correlation. The "Lena" and "Bamboo" with size 128×128, resized by Adobe PhotoShop CS2, were utilized. The random generated watermark was encoded by these three codes, and 10 different watermarks were applied to each code.

Parameter α, which is the strength of the embedded watermark as used in [10], was tested in the range 0.02–0.06, as shown in Figs. 4-5. Experimental results demonstrate that 438 words could be embedded into one image. The number of error words approached zero with increasing α. Many more error words appeared in "Bamboo" than in "Lena" with small α, probably because "Bamboo" has a large high-frequency component. BIBD performed better than random generation coding and orthogonal coding with small α, and approached 0 with increasing α. The results were obtained from 10 different watermarks and the computation of the average error words number in each α and image. All the random generated code and orthogonal code were the same size as the BIBD code, and were decoded by the correlation approach.

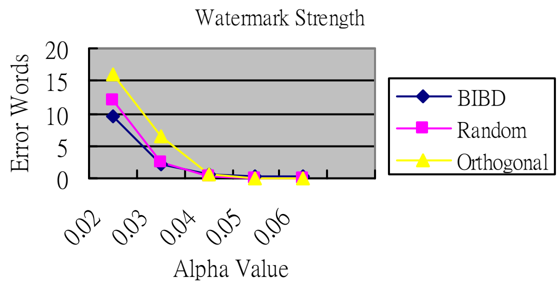

Fig. 4 Number of error words extracted by three coding approach in "Lena"

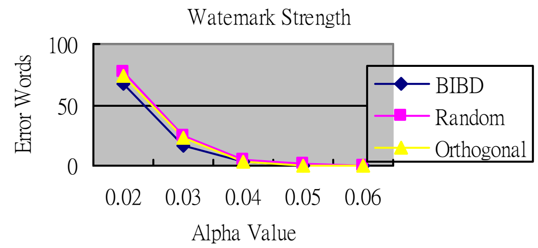

Fig. 5 Number of error words extracted by three coding approach in "Bamboo"

After testing the error words, Gaussian distribution noise (using function "normnrnd" in Matlab version 7.01) was added to the watermarked image "Lena", setting $\alpha = 0.05$ and 0.04. The standard deviation of the noise was 5 to 1, and the noise was with zero mean. Figures 6–7 show the analytical results, which show that the BIBD coding and random coding was more robust than orthogonal coding approaches to Gaussian noise attack when the deviation was high. (The value of result is for reference only, since the noise was added randomly, and was thus different each time it was added.) In this experiment, each deviation was run 100 times at different noise levels. Figures 3.4–3.5 demonstrate the average results of error words.

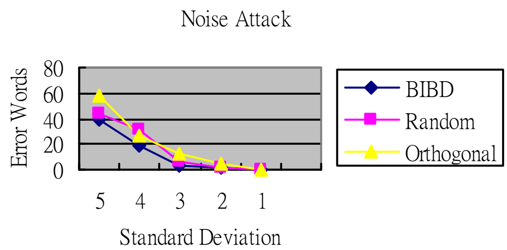

Fig. 6 Number of error words after Gaussian noise attacked ($\alpha = 0.05$)

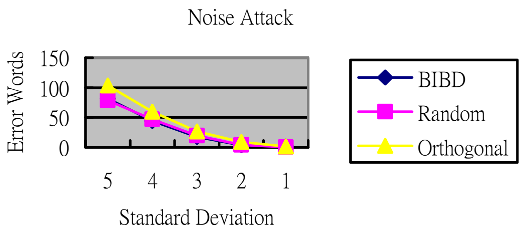

Fig. 7 Number of error words after Gaussian noise attacked ($\alpha = 0.04$)

Then the experiment utilizes "Lena" with sizes 128× 128 (Fig. 8(a)) to test the robustness to cropping attack. The watermarked image (Fig. 8(b)) is cropped as shown in Fig 8 (c)~(g), using rectangles with sizes from 50×50 and 40×40, from the left and top side to the right and bottom side. Experimental results demonstrate that the proposed approach is robust to cropping attack. The error word was close to zero when the cropped size was small as shown in Fig. 9..

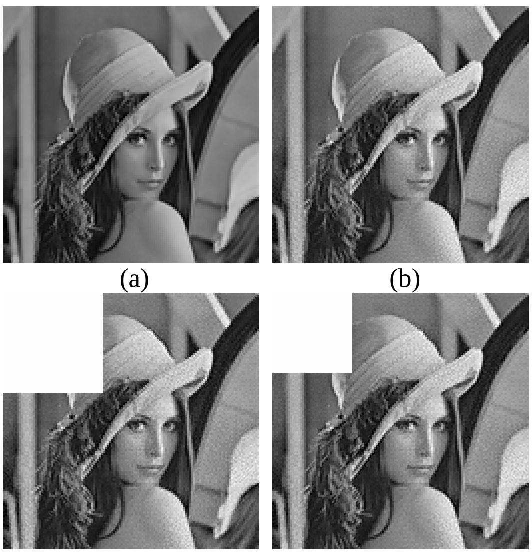

Fig. 8 "Lena" image (a) original, (b) watermarked with $\alpha = 0.05$, (c) 50×50 cropping, (d) 40×40 cropping.

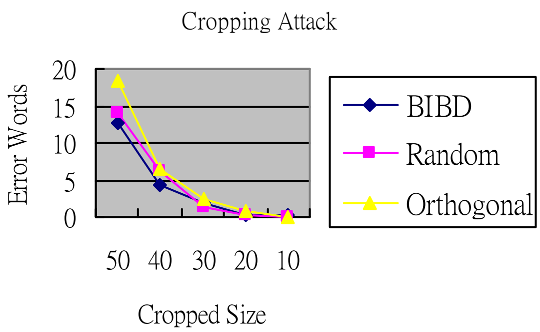

Fig. 9 Number of error words in cropping attack in image Lena.

The final experiment tested the robustness against JPEG compression. The JPEG compression in this experiment was performed using the function "imwrite" in Matlab version 7.01. The "Quality" parameter was set as 50–100%- the "Lena" image with size 128×128 was used, and the watermark was added by the three coding approach discussed herein with $\alpha = 0.05$. Figures 10-11 presents the experimental results. The watermark was added to the DWT coefficient at level 0 (high frequency component). Embedding the watermark into level 1 or even level 2 might improve the robustness to JPEG compression, but would also decrease the invisibility of the watermark.

Fig. 10 Watermarked "Lena" image after JPEG compression with Quality = 80%

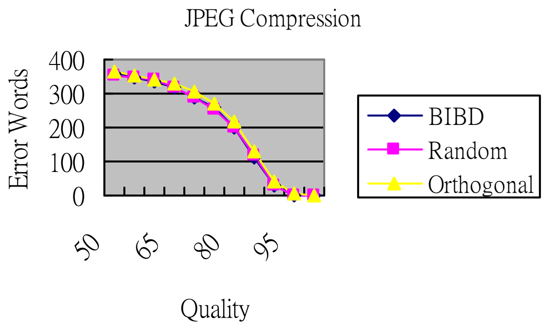

Fig. 11 Number of error words of three coding approach by JPEG compression

4 Conclusion

A watermarking approach has been proposed to use the mathematical structure BIBD to design the watermark. Watermark embedding is achieved by embedding the watermark into the detailed coefficients in the wavelet domain. Perceptual analysis is also utilized to increase the watermark strength.

According to the experimental results, the proposed approach is reasonably robust to Gaussian noise, cropping, and JPEG compression. By utilizing the mathematical structure of BIBD, the size of embedded watermark is small and the capacity of the embedded information can be enhanced.

Future work may focus on taking advantage of the other characteristics of BIBD such as the parameter *r* to improve the performance of the watermarking system. And the error correction scheme may be important in some applications which put emphasis on the correctness of the message such as military information exchange. For the DWT watermarking, we can design a scheme to avoid the error arise from the HVS analysis values equal to zero.

5 References

[1] I. Cox and M. L. Miller, "A Review of Watermarking and the Importance of Perceptual Modeling," in *Proc. Electronic Imaging*, Feb. 1997.

[2] X. Xia, C. G. Boncelet, and G. R. Arce, "A Multiresolution Watermark for Digital Images," in *Proc. 4th IEEE Int. Conf. Image Processing '97,* Santa Barbara, CA, Oct. 26-29, 1997, pp. 548-551.

[3] D. Kundur and D. Hatzinakos, "A Robust Digital Image Watermarking Method Using Wavelet-based Fusion," in *Proc. 4th IEEE Int. Conf. Image Processing '97*, Santa Barbara, CA, Oct. 26–29, 1997, pp. 544–547.

[4] C.-T. Hsu and J.-L. Wu, "Multiresolution Watermarking for Digital Images," *IEEE Trans. Circuits Syst. II*, Volume. 45, pp. 1097–1101, Aug. 1998.

[5] D. Kundur and D. Hatzinakos, "Digital Watermarking using Multiresolution Wavelet Decomposition," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, volume 5, Seattle, WA, May 1998, pp. 2969–2972.

[6] W. Zhu, Z. Xiong, and Y.-Q. Zhang, "Mutliresolution Watermarking for Images and Video," *IEEE Trans. Circuits Syst. Video Technol.*, volume 9, pp. 545–550, June 1999.

[7] M. D. Swanson, B. Zhu, and A. H. Tewfik, "Multiresolution Scene Based Video Watermarking Using Perceptual Models," *IEEE J. Select. Areas Commun*., volume 16, May 1998.

[8] A. S. Lewis and G. Knowles, "Image compression using the 2-D wavelet transform," *IEEE Trans. Image Processing,* vol. 1, pp. 244–250, Apr. 1992.

[9] D. Vleeschouwer, C., J.-F. Delaigle, and B. Macq, (2002) "Invisibility and Application Functionalities in Perceptual Watermarking－An Overview," *Proc. IEEE*, Volume 90, No. 1, pp. 64-77.

[10] M. Barni, F. Bartolini, and A. Piva, "Improved Wavelet-based Watermarking Through Pixel-wise Masking," *IEEE Trans. Image Process.*, volume 10, pp. 755-766, May 2001.

[11] C. J. Colbourn and J. H. Dinitz, The CRC Handbook of Combinatorial Designs, 1996.

[12] I. J. Cox, M. L. Miller, and J. A. Bloom. *Digital Watermarking* Academic Press, San Francisco, 2002.

[13] J. F. Delaigle, C. D. Vleeschouwer, and B. Macq. "Watermarking Algorithm Based on Human Visual Model." *IEEE Transactions on Signal Processing*, Volume 66(3):319–335, 1998.

[14] J. M. Barton. "Method and Apparatus for Embedding Authentication Information Within Digital Data," *United States Patent* 5,646,997, 1997.

[15] R. B. Wolfgang and E. J. Delp. "A Watermark for Digital Images," in *Proceedings of the 1996 International Conference on Image Processing*, volume 3, pp. 219-222, 1996.

[16] B. A. Wandell, *Foundations of Vision.* Sunderland, MA: Sinuaer, 1995.

[17] B. Girod, "Psychovisual Aspects of Image Communication," *Signal Processing*, volume 28, no. 3, pp. 239-252, Sept. 1992.

[18] L. A. Olzak and J. P. Thomas, *Handbook of Perception and Human Performance: Seeing Spatial Patterns*. New York: Wiley, 1986, volume 1, chapter 7, pp. 1-56.

[19] J. L. Mannos and D. J, Sakrison, "The Effects of a Visual Fidelity Criterion on the Encoding of Image," *IEEE Trans. Inform. Theory*, volume IT-20, pp. 525-536, July 1974.

[20] C.R. Carlson and R. Cohen, "A simple psychopysical model for predicting the visibility of displayed information," *Proc. Soc. Info. Display*, volume 21, pp. 229-245, 1980.

[21] I. J. Cox and M. L. Miller, and A. L. McKellips, "Watermarking as communication with side information," *Proc. IEEE*, volume 88, pp. 1127-1141, July 1999.