# Embedding a Watermark in Multimedia Archives

Apostolos C. Klonis
Informatics Secondary school Teacher, M.Sc., PhD.,
apoklonis@gmail.com
atklonis@itl.auth.gr

**Abstract** It is usual to come across types of images on the Internet that have characteristics which correspond to their nature as a medium. So the most common formats are Tagged Image File Format TIFF (TIF), Graphics Interchange Format (GIF), Joint Photographic Experts Group (JPEG or JPG) and Portable Network Graphics (PNG). This article will explore the ways in which an image in a compressed or uncompressed form can be shared in a collaborative, online artwork. Obviously, sharing must be done with the aim of protecting the data from malicious attacks and at the same time ensuring the quality of the work.

**Index Terms**— Image, attack, watermark, multimedia, integration..

## 1. Introduction

It is usual to come across types of images on the Internet that have characteristics which correspond to their nature as a medium. So the most common formats are Tagged Image File Format TIFF (TIF), Graphics Interchange Format (GIF), Joint Photographic Experts Group (JPEG or JPG) and Portable Network Graphics (PNG).

## 2. Multimedia Archives

### 2.1 Tagged Image File Format TIFF (TIF)

This template is suitable for creating high quality digital images. The corresponding files support compression without loss of information or are stored without compression, thus occupying a large volume. Each scanner and digital camera can generate TIFF files either directly or as an option to export the image to the software that came with the device.

According to the digitization guidelines, TIFF is considered to be the most suitable for storing digital copies, unless there are serious and sufficient ones that dictate the choice of a different file type. The current version of TIFF is 6.0. This file type does not alter the quality of the images. The files have excellent display quality but are large in volume. Usually, they are used in professional typography.

### 2.2 Graphics Interchange Format (GIF)

Graphics Interchange Format (GIF) is the method of compressing and / or encoding graphics on computers, as well as the corresponding file type (format) of the image produced through it. Developed by the American company CompuServe and introduced in 1987, it was the most common type of graphics for several years. There are two versions of GIF files: the original GIF87a version of 1987 and the newer enhanced version of GIF89a introduced in July 1989. It supports up to 256 colors, with 1-8 bits per pixel, as well as the ability to store multiple images in the same file. GIF compression is done using the Lempel-Ziv-Welch (LZW) algorithm.

The fact that this method was until 2003/4 a patent eventually granted to Unisys, was the cause of its legal dispute with CompuServe, when since 1994 Unisys demanded a fee for licensing the use of GIF files. This type only supports 256 colors, it is suitable for applications that do not require much detail but require a small file size. It supports animated graphics (as multiple images in the same file create a sense of motion - category: animated gif) as well as the ability of a transparent background that serves us in the design of web pages.

### 2.3 Joint Photographic Experts Group (JPEG or JPG)

JPEG is a standard image loss compressor created by the Joint Photographic Experts Group-JPEG from which it took its name. Due to the small file size that can be obtained with this compression method, it is mainly used in web pages (as well as GIF) and in cameras: in high resolutions an image that has not been compressed can use up to 40MB of space while in JPEG uses about 3MB. JPEG file extensions are .jpg .jpeg .jif .jpe .jfif. It is the standard that one encounters mainly on the internet and supports up to 16.7 million colors. Images saved as .jpeg or .jpg have been compressed to reduce the size of the file. If an image is saved in this format, its details cannot be retrieved and the image quality is reduced. Because JPEG involves lossy compression, image defects appear. Depending on the compression level selected (0 to 100) the image quality increases or decreases along with the file size.

There are different types of imperfections in the JPEG image. One drawback is the division of the image into 8x8 pixel squares. This phenomenon is called "macroblocking". Other imperfections are chromatic aberration, distortion of the edges of the image and non-uniformity of colors (colors are not solid and are mixed at the edges of the object being depicted). However, these losses are often not visible if the display is on a computer screen. They start to appear when the compression ratio is large enough (more than 60%) and the display is done through a video projector on a large screen. JPEG images are not suitable for use in printing jobs in printers or large plotters and their use in these cases is avoided [2].

### 2.4 Portable Network Graphics (PNG)

PNG was developed to overcome the limitations of GIFs. PNG works in two color ranges, 256 (8 bit) and 16cm (24 / 32bit). On the Web, 8bit is more interested in saving space. PNG images can be 5% -25% more compressed than GIFs, while at the same time offering more control over transparency through the ability to adjust opacity. The main problems of PNG are

ISSN 2229-5518

the incompatibility with some browsers and the absence of animation, something that keeps the GIFs still alive. In general, they are characterized by good quality combined with a small file size but larger than JPEG and GIF. In the "battle" between formats is the bet of the future.

## 3. Implementation Stage

### 3.1 Watermark Integration

Given a set of particle coefficients, it has been observed that the average has a smaller variation than an atomic coefficient. Thus, the watermark embedded in the average of the particle segments is more powerful than at any coefficient. In this section, the deep-wavelength coefficient is selected as the carrier of the watermark signal. The matrix of the deep-wavelength coefficient of a Ca can be achieved by 2D wavelet decomposition. M and N are the length and width of the particle block respectively and Ii (k) is the number i particle coefficient in the number k particle block. Of course, in the variable i [1, M × N] the n LSBs of Ii (k) are defined as [3],

(k) = mod (Ii (k), 2n) (1)

Next, the average of the particle blocks is defined as follows.

$$Average(k) = \frac{\sum_{i=1}^{MxN} \hat{I}(k)}{MxN} \quad (2)$$

If some of Ii (k) have changed from Ω due to some distortion, the average particle block will have only a small change. Assuming that I'i (k) is the no. i particle coefficient in k particle block after embedding the watermark, I ′(k) is the n LSBs of′ (k) and is its average, ′(k) in number k particle block respectively. The watermark W, consisting of a binary pseudo-random series, W (k), is integrated by adjusting the average of the block particles in this way

$$Average'(k) \in \begin{cases} [0, 2^{n-1}), if W(k) = -1 \\ 2^{n-1}, 2^{n}), if W(k) = 1 \end{cases} \quad (3)$$

As shown in Figure 1, there is a maximum distance between integers 2n-2 and 2n-2n-2

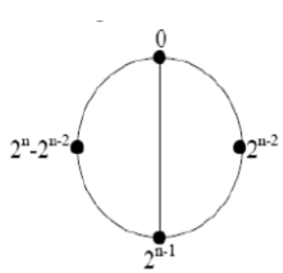


Fig 1: The n bit distance in the mod 2n cycle.

Prior to adjustment (k), there are three outstanding issues. First, whether or not each individual factor n LSBs needs adjustment, and if not, how do you note it? Second, how do you draw the embedded strength if (k) has satisfied equation 3 before any adjustment is made? Third, how do you use optical models when adjusting individual n LSBs?. These issues will be discussed in the following text. In the first sections on DCT-based watermarks, each particle coefficient is adjusted to make the average equal to 2n-2 or 2n-2n-2. Therefore, an identifier function must be configured to decide whether i (k) needs adjustment or not. The marking rule is defined as follows

Fi(k) =sign((2n-1-$\hat{I}$ i(k))xW(k)) (4)

$$\text{When sign}(x) = \begin{cases} 1, if x \geq o \\ 0, if x < 0 \end{cases}$$

The detailed result of the operation of the identification point is shown in Figure 2.

| $W(k)$ | $2^{n-1} - \hat{I}_i(k)$ | $2^{n-1} - Average(k)$ | $F_i(K)$ | $S(K)$ |
|---|---|---|---|---|
| -1 | >0 | >0 | 0 | 0 |
| -1 | ≤ 0 | ≤ 0 | 1 | 1 |
| 1 | >0 | >0 | 1 | 1 |
| 1 | ≤ 0 | ≤ 0 | 0 | 0 |

Fig 2: Results of operation of identification points

In addition, it is possible that (k) has satisfied Equation 3 before any adjustment is made. Thus, in this case, no adjustment is needed on these particle blocks. Given the robustness of the watermark demonstrated in the above paragraphs, a very slight adjustment is still necessary. Otherwise, if the average k is not a satisfactory equation3, a relatively strong adjustment is required [1]. Thus, an estimate between 2n-1 - average (k) and W (k) to 1determine the adjustable force is necessary when the watermark is integrated. The power function S (k) is defined as

S (k) = sign (x (k)) (5)

Where x (k) = (2n-1- Average (k)) xW (k)

The detailed result of the power operation is shown above. To adapt the watermark sequence to the local properties of the particle block, we use the HVS-based model on the watermarked system. The model is similar to that of reference, but has been developed independently. The optical model takes into account the light sensitivity and the texture sensitivity of the particle blocks to noise. The function of the optical model Vm (k) is defined as:

Vm(k)=brightness(k) χ texture(k)β (6)

Where: texture(k)=, $\frac{\sum_{i=1}^{MXN}[brightness(k)-Ii(k)^2]}{MXN}$

brightness (k)= $\frac{\sum_{i=1}^{MXN} Ii(k)}{MXN}$

and b to control the degree of texture sensitivity. This visual model function shows that the human eye is less sensitive to noise in very bright areas of the image. Therefore, particle blocks are divided into two parts that depend on the value Vm (k): high-activity particle blocks and low-activity particle blocks. For the sake of simplicity, the limit Tc is set for the average of Vm (k) [2]. The following function can be applied to distinguish high or low activity particle blocks.

T(k)=sign(Vm(k)-Tc) (7)

Taking into account the balance between robustness and transparency, the proposed algorithm that incorporates a watermark can be formulated as follows.

$\hat{I}$ i(k)= $\hat{I}$ (k)+aW(k)Fi(k)[2n-2-S(k)+T(k)x2n-3] (8)

where α is an incremental coefficient to check the strength of the imported watermark. Based on the above discussion, the n LSBs of the particle coefficients are adjusted using equation 8. Of course, their average has been updated according to the requirement W (k), as shown in equation.3. In other words, the watermark is embedded.

## 4. INTEGRATION OF SINGLE WATERMARK SWE (SINGLE WATERMARK EMBEDDING)

In this article, a SWE system is proposed to integrate a watermark into an image. SWE can be applied to conversion areas such as DCT and possibly to the spatial area of an image. Some selected image pixels or conversion factors are grouped to form a vector and name the host of the watermark. Suppose that the host of the watermark Y = [y1.y2 …… .yM] is of length M .The watermark L = [l1.l2… lN] of li {0.1} is a bit sequence of length N, where N < M [3]. The bit sequence can be a common image, such as the logo of the owner image or the information associated with the host images, such as the owner name, the ID image, etc. The watermark is differentiated by a logical XOR operation together with a pseudo-random bit sequence S = [s1.s2… .SN] with Si {0.1}, to give the configured watermark sequence W = [w1.w2… .wN] with wi = si li. In the rest of the section, W will simply be used to represent the configured watermark. To incorporate the watermark, divide the host Y into N subunits of equal duration P = [M / N] with the i subcarrier denoted as Yi.

Each carrier is used to incorporate a bit of watermark information. The SWE uses two keys to integrate W into the host vector Y, to form the watermarked vector Y'i. The first key, denoted D = [d1.d2… .dN] di R + i N] is a set of pseudo-random N positive real numbers. The second key is K = [k1.k2… .kM] with every ki being zero - Gaussian with mean variation σi. Both keys are needed to decode or locate the configured watermark. Similar to the host Y the two parts K and Y'I are divided into N subunits of equal duration P with the i subcarrier being denoted as Ki and Y'i respectively [4]. The axis of Ki is divided into discontinuous cells of width di and the cells are assigned alternately with "1" and "0". The watermark i bit is integrated by adding a slight deviation in the direction of Ki. Y'i = Yi + aiKi (1), where in equation (9):

$$(9)ai=\begin{cases} \frac{di * round(\frac{<Yi,Ki>}{di}) - <Yi,Ki>}{\|Ki\|_2^2}, \\ \frac{di * (round(\frac{<Yi,Ki>}{di}) + 1) <Yi,Ki>}{\|Ki\|_2^2}, \\ \frac{di * (round(\frac{<Yi,Ki>}{di}) - 1 <Yi,Ki>}{\|Ki\|_2^2}, \end{cases}$$

case 1: $round(\frac{<Yi,Ki>}{di})\%2 = w_i$

case 2: $round(\frac{<Yi,Ki>}{di})\%2 \neq w_i$ and $<Yi,Ki> \geq d_i * round$ $(\frac{<Yi,Ki>}{di})$

case 3: $round(\frac{<Yi,Ki>}{di})\%2 \neq w_i$ and και $<Yi,Ki> < d_i * round$ $(\frac{<Yi,Ki>}{di})$

where round (),%, and || * || 2 are the roundings, coefficients 2 and L2 standard respectively, and <Yi.Ki> is the inner product of Yi and Ki In case 1, the projection of Yi on the axis Ki falls into a cell of the desired watermark bit. In cases 2 and 3, the view is in the wrong cell. In case 2, the cell on the right is closer. In case 3, the cell on the left is closer. The projection of Y'i is enhanced to be in the center of the nearest cell of the desired watermark bit. The nearest cell will ensure minimal distortion of the image. The center of the cell will give maximum strength, because the distance to the nearest cell limit is the maximum at the center of the cell. It would take a distortion of at least di / 2 in the direction of Ki to charge an error in the i decoded bit. In this sense, the di cell width controls the robustness of the built-in watermark. However, a large di can result in visible objects as the watermark distortion is greater than di / 2 in cases 2 and 3. To decode the watermark bits for the SWE, the watermark vector Y 'is extracted from the test image and separated in N conifers of length P [5]. The i configured watermark bit w'i is decoded by the i subcomponent Y'i as w'i = round (<Y'i.Ki> / di)% 2 using the two keys and the I demodulated watermark bit is taken as l'i = w 'si. To determine if a configured watermark W = [w1.w2… .WN] is present in a test image, all N bits of the watermark from the image have been decoded as W = [w'1.w'2… .w ' N] and a result is evaluated. One possible result is the traditional normalized S1 intersection correlation between the original watermark and the decoded watermark in equation (10).

$$\frac{\sum_{i=1}^{N}(2*wi-1)*(2*w'_i-1)}{\sqrt{\sum_{i=1}^{N}(2*wi-1)^2*\sum_{i=-1}^{N}(2*w'i-1)^2}} = \frac{1}{N}\sum_{i=1}^{N}(2*wi-1)*(2*w'i-1) \quad (10)$$

Each bit of watermark is converted from {0,1} to {-1,1} in (10). When SWE is applied to the conversion domain, one possible result is the weighted normalized cross-correlation S2=

$$\frac{\sum_{i=1}^{N}[\beta i*(2*wi-1)]*[\beta i*(2*w'_i-1)]}{\sqrt{\sum_{i=1}^{N}[\beta i*(2*wi-1)]^2*\sum_{i=-1}^{N}[\beta i*(2*w'_i-1)]^2}} = \sum_{i=1}^{N}\frac{\beta_i^2}{\sum_{j=1}^{N}\beta_i^2}*(2*wi-1)*(2*w'i-1) \quad (11)$$

where βi>=0. Relatively large bi is selected for the watermark bits embedded in the low-frequency elements of the image and larger bi is selected for the high-frequency elements, because the decoded bits in the low-frequency elements tend to be more reliable. Attacks (intentional or unintentional) on low-frequency components tend to be less powerful than on high-frequency components, because distortions on the low-frequency components tend to be more visible. If the detection result is greater than a predetermined limit, the watermark is considered to be present in the test image [10].

## 5. M.W.E. (MULTIPLE WATERMARK EMBEDDING) INTEGRATION

In this section, SWE will be generalized to integrate multiple watermarks into the same image while maintaining high image quality. In SWE, only one bit is embedded in each subcarrier. In MWE, Q bits are integrated simultaneously into each Yi subcarrier. Thus, the first key for Q sets N of pseudo-random positive real numbers denoted as D1.D2… ..DQ is generalized. And the second key in Q pseudo-random carriers of length M declared as K1.K2… .KQ with Kj = [kj.kj2… kjM] for1. Similar to SWE, the vector is divided host Y and the random carriers K1.K2… .KQ in N subductors of length P. The i element ofDj is denoted as dji and the i subunit of Kj is denoted as Kj, i. The i bit of the j watermark sequence is denoted as wj, i. The watermarked i carrier, denoted as Y’i is [11]:

$$Y'i=Yi+a1.iK1.i+\alpha2.iK2.i+\ldots\ldots+\alpha Q.iKQ.i \quad (12)$$

## 6. I.W.E. (INTERACTIVE WATERMARK EMBEDDING) INTEGRATION

An interesting problem arises when the original image for the proposed watermark algorithm is a JPEG - a compressed image from one .jpg file and the watermark image must be JPEG recompressed to produce another. jpg file. Can the watermark be decoded or detectable in the JPEG recompressed file? Since the original image is JPEG-compressed, the original DCT factors are already quantized. For any watermarking method, the distortion of these quantized DCT coefficients due to watermarking is small compared to the quantization coefficient. This is especially true when the compression ratio is high. Based on SWE, we propose a new IWE method to prevent the watermark from being removed in the re-quantization process so that the watermark decoding and detection can still function. IWE assumes that the watermarked image will be recompressed with the same quantization matrix and quantization factor as the original JPEG - compressed image. Initially, the M-length host vector watermark is extracted directly from the JPEG compressed region of the original image. To form the host vector watermark, the first AC factor in the zigzag series is subtracted from the set of 8x8 squares to form the first part of the Y. Then the second AC factor in the zigzag series is subtracted from the squares and is attached to form the second part of Y, and so on, until a set of M factors are extracted. We split the host Y into N subunits of P. Here is the proposed IWE for each sub carrier Yi : for any Yi, a random noise carrier Ni of length P is generated and added to Yi before SWE is applied. If the initial quantized DCT coefficient is zero, the random noise component is required to be zero [7]. After embedding the watermark, the JPEG quantization is performed followed by the decoding of the watermark. If the bit (or bits) of the embedded watermark can be decoded correctly, the target is reached and IWE stops [8]. Otherwise, another iteration is performed with another random Ni noise carrier and the other steps are repeated. To reduce complexity, the iteration is forced to end when the number of consecutive approaches exceeds a predefined limit.

## 7. BLIND WATERMARK SYSTEM FOR DWT - BASED IMAGES

With the explosive growth of the Internet in recent years, the protection of the copyright of multimedia data is becoming one of the most important issues in the world of the Internet. As a dynamic and effective way to solve this problem, digital watermark is becoming a very active area of signal research and information processing. A digital watermark is a signal that is embedded in a digital image or video sequence. It will allow someone to determine ownership, identify a buyer, or provide some additional information about digital content. In general, it is required that the embedded watermark should not only be transparent to human observers, but also strong enough that it cannot be damaged or removed after any treatment or attack. In general, the process of embedding the image watermark for digital images can be accomplished in either the spatial or frequency domain. It has been shown that a better compromise between robustness and transparency can be achieved by using the frequency domain system [11]. Frequency domain watermarking techniques could be based on spatially local or global transformations. In the existing literature, a famous frequency domain regime for digital watercolor images has already been proposed by Cox. One of the important contributions to this project is the realization that the watermark must be integrated into the important part of the image in order to be robust [9].

## 8. Conclusions

### 8.1 Selection and Justification of an Appropriate Algorithm for Digital Image

From the previous study, it is argued that the best method for digital image watermarking that will be distributed on the Internet is blind systems based on discrete cosine transformations (DCT). Unlike most watermarking systems, the watermark is not integrated by adjusting the coefficients of the individual particles, but by adjusting the average coefficients in the particle blocks. Detection of the watermark is not complete without the original. The results of simulations made by various studies show that the proposed regime is transparent and robust in the processing of many common images, such as compression, noise addition, scaling, cropping, rotation, filtering and more. In the pictures below we can see the results [10].



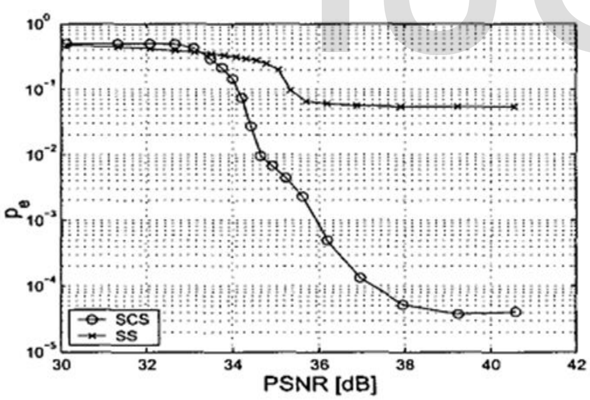


Fig 3: Experimental results

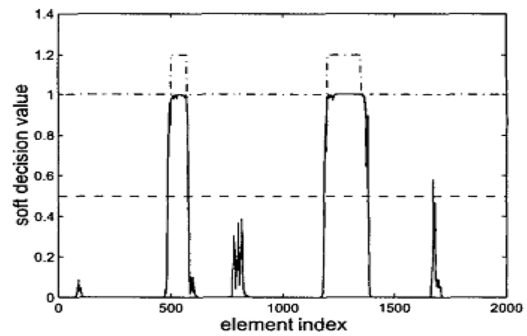


Fig 4: Experimental results

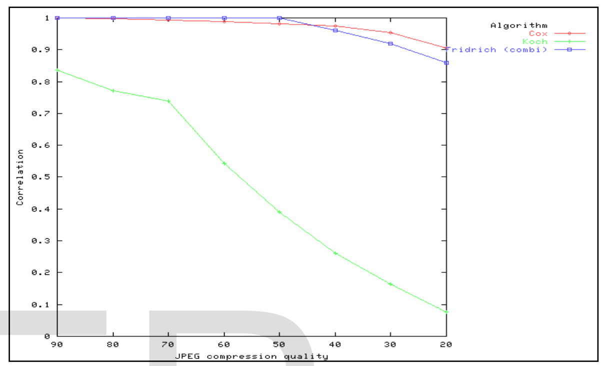


Fig 5: Blind algorithms

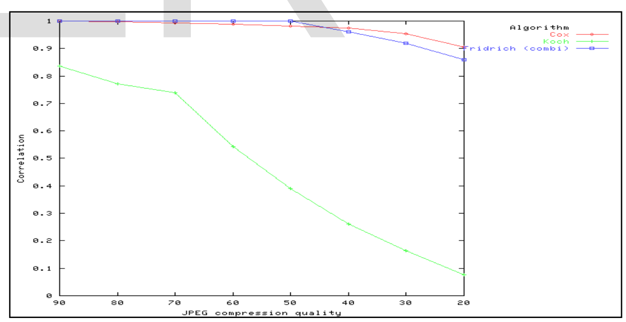


Fig 6: DCT algorithms

## 9. Acknowledgment

The author would like to thank teachers who give knowledge about how to read papers and write papers.

## 10. References

[1] B. Chen, G. Wornell, "Quantization Index Modulation: A Class of Provably

Good Methods for Digital Watermarking and Information Embedding," IEEE Trans. Information Theory, vol. 47, no. 4, pp. 1423-1443, May 2001.
[2] B. Chen, G. Wornell, "Quantization Index Modulation Methods for Digital Watermarking and Information Embedding of Multimedia," Journal of VLSI Signal Processing, no. 27, pp. 7-33, 2001.
[3] B. Friedrichs. Kanalcodierung (in German), Springer, Berlin, 1996.
[4] B.P.Lathi, "Modern Digital and Analog Communication System", Oxford University Press, 1998, pp.728-737.
[5] Cox.I.J, Kilian.J, Leighton.T, Shamoon.T, "Secure spread spectrum watermark for multimedia," Proceedings of NEC, Princeton, USA, pp.95-100, Oct.1995.
[6] C. I. Podilchuk, E. J. Delp, "Digital watermarking: algorithms and applications,"IEEE Sig. Proc. Mag., vol. 18, no. 4, pp. 33-46, July 2001.
[7] C.I. Podilchuk, W. Zeng, "Image-Adaptive Watermarking Using Visual Models," IEEE Trans. Selected Areas of Comm., vol.16, no.4, pp.525-539, May 1998.
[8] [C.P.Wu, P.C.Su, C.J.Kuo, "Robust Audio Watermarking for Copyright Protection", in Proceedings of SPIE's 44th Annual Meeting on Advanced Signal Processing Algorithms, Architectures, and Implementations IX (SD39), 1999.
[9] C.-T. Hsu, J.-L. Wu, "Hidden digital watermarks in images," IEEE Transactions on Image Processing, vol. 8, no.1, pp. 58-68, 1999.
[10] D. Kundur, D. Hatzinakos, "A robust digital image watermarking method using wavelet-based fusion," Proc. 4th IEEE Int. Conf. Image Processing, ICIP '97, vol. 1, pp.544-547, Santa Barbara, CA, USA, Oct. 26-29, 1997.
[11] D. Kundur, D. Hatzinakos, "Digital watermarking for telltale tamper proofing and authentication," in Proc. IEEE Special Issue on Identification and Protection of Multimedia Information, vol. 87, no. 7, pp. 1167-1180, July 1999.