A Unique Watermark for Every Image

Andrew Z. Tirkel and Thomas E. Hall
Monash University, Australia

We address image security issues by proposing a scheme for watermarking images or video at the camera. Each watermark is unique and identifies the owner, camera, and frame number. Application areas include the security industry (for proof of tampering) and commercial photography (for copyright protection).

Digital image watermarking has escalated in relative importance because of the Internet's proliferation. Since our research team at Monash University introduced the digital watermark[1] in 1993, significant improvements in the technology have occurred.[2,3] Almost all the new techniques involve inserting a watermark into image data. Most of the issues relating to a watermark's robustness and extraction involve transmitting it from one computer to another, leaving it vulnerable between its point of origin and the computer where someone inserts a watermark.

We propose a dedicated spatial pseudonoise watermark to protect data between its points of origin and distribution. We achieve watermark recovery by performing a correlation of the watermarked image with the watermark template. Insertion in the spatial domain is preferable because that's how the captured data first appears at a charge-couple device (CCD) sensor's output. In principle, we could apply the scheme to a transformed image, but that would leave the image vulnerable to tampering before the transform.

In our scheme, no unwatermarked image exists except its estimate after we remove the watermark. This watermark can be compatible with the acquisition hardware of standard CCD cameras, and we can insert it at the analog–digital converter stage. Its implementation in an electrically programmable logic device (EPLD) is straightforward and only requires basic building blocks such as counters, shift registers, and adders.

Table 1. Result table without an original image.

Test	$W_A \in I_B$	$W_A \notin I_B$	Inconclusive
$W_A \in I_B$	Both	B	B?
$W_A \notin I_B$	A	Neither	Inconclusive or tentative
Inconclusive	A?	Inconclusive or tentative	Inconclusive or tentative

Our watermark can coexist with the standard watermarks used for protecting images during distribution. Traditional watermarks are usually implemented in a transform domain and are optimized for robustness against compression and geometrical distortions (such as rotation, scale manipulations, cropping, and shear). Because compression standards are evolving, a distribution watermark requires flexibility. It's impractical to use one watermark to perform these seemingly incompatible functions. Different compression methods (and consequently distribution watermarks) also impact the watermark's effectiveness in resolving proof of piracy or tampering issues. For example, fractal compression is radically different in concept and application to discrete cosine transform based compression, as are the watermarking schemes for these two cases. Researchers have yet to resolve robustness of one scheme to attacks based on another method. An attacker can use fractal compression and decompression to generate telltale artifacts deliberately to obfuscate the process.

Lastly, with our spatial pseudonoise watermark construction method, we can provide individual watermark patterns for each image. We also discuss embedding and recovering multiple watermarks from one image, which enhances the information capacity of such watermarks but renders them more vulnerable to errors and deliberate attacks.

Watermarking at the point of origin

In commercial photography, a watermark at the point of origin can resolve conflicts pertaining to image piracy. Consider a situation where A claims that B has pirated her image I_A on the basis that B's image I_B contains A's watermark W_A and therefore B's image is a copy. B might have a similar claim based on the watermark W_B. We can interpret tests—supervised by an independent authority—involving each watermark's detection by consulting a result table. For example, see Table 1, where we translate detection of A or B's watermark in each of the two images into who is a pirate. Note that there is no reference to an original, unwatermarked image nor is

1070-986X/01/$10.00 © 2001 IEEE

this a proof that A or B can rightfully claim that original.

In schemes where an original is required for detection or recovery, our table becomes complicated because we must test four objects that might or might not be watermarked. This is in spite of the fact that all the claimants can produce a self-consistent method of extracting their watermarks.

Clearly, the former method is more effective. However, successfully detecting or recovering a watermark, without using an original, places more restrictions on the watermark's design, especially in terms of its cross-correlation with the image and other watermarks. It's desirable to implement such a scheme by applying the watermark from a single, in-line memory module (SIMM) card inserted into a digital camera (see Figure 1), in a similar manner to that in cellular telephones. We can incorporate the hardware required to insert the watermark into an EPLD at the CCD's output, before any image processing takes place.

The security industry has implemented video surveillance on an unparalleled scale. These cameras can acquire huge numbers of video frames and multiplex them into a common storage data bank. Ensuring that such data is admissible evidence in a court of law requires some quantitative proof of integrity. This involves proof of tampering with an individual frame, frame sequencing, or synchronizing the sound track.

Typically, we should establish tampering with a frame on the basis of upper and lower thresholds. Combining a digital signature (fragile) with a correlative pseudonoise watermark can achieve this objective. We can address sequencing and synchronization issues by assigning each frame a unique watermark with low cross-correlation with all others.

Watermark construction

Our original spatial watermark construction method[1] involved sequences developed for spread-spectrum communications. Since then, it has become clear that, despite many common features, watermarking requirements are different because of the following reasons:

- Images and video are multidimensional. This results in causal relationships between pixels in more than one dimension. Researchers have constructed and analyzed multidimensional arrays[4] and applied them to coded aperture optics and structured light, but there, a single array is sufficient.

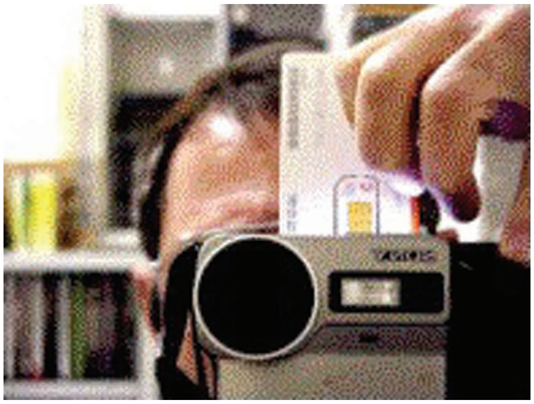

Figure 1. Watermark via a SIMM card in a digital camera.

- The image data isn't embedded on a (sinusoidal) carrier. This makes it difficult to embed watermarks based on roots of unity, as we do in communications with phase modulation. Our group has recently demonstrated two schemes that achieve this, for color and grayscale images, by constructing angles in abstract spaces.[5] (Svalbe, Tirkel, and van Schyndel[6] give an analysis of such methods.)

- The pixels are quantized and image polarity is usually known. Therefore, correlation is meaningful for integer pixel offsets only and coherent statistics are involved. Hence, we should consider correlation a signed quantity. This contrasts with spread-spectrum communications and produces different criteria for acceptance of arrays suitable for watermarking. It also has implications when more than one watermark is involved, with watermarks and correlations adding arithmetically.

Watermark properties

For watermarking schemes relying on correlative recovery, the most important watermark requirement is that it should be imperceptible but still yield an unambiguous autocorrelation peak. Ambiguities can arise because of cross-correlation with the image, autocorrelation sidelobes, and cross-correlation between different watermarks. We can reduce the effects of the cross-correlation with the image with nonlinear (median) filtering before correlation,[4] or with Laplace filtering. These techniques rely on the image's different spatial structure from that of a pseudonoise sequence or array. We can reduce the cross-correlation between different watermarks with orthogonalizing filters.[7]

However, such filters result in a small family of watermarks with degraded autocorrelation peaks.

Cross-correlation with the image affects filter design and stability. Both techniques are useful adjuncts to a good watermark design. Therefore, we've searched for large families of watermarks with good autocorrelation and cross-correlation. An added benefit of such a design is that we can embed multiple watermarks in the same image to increase information capacity, enhance robustness, or decrease visibility in the image's low-contrast regions. Many other, more flexible watermark requirements exist for video and still images, such as image size, aspect ratio, robustness against geometrical distortions, compression, and cryptographic attack.[3,4]

Array constructions

Computers have done exhaustive searches and assisted searches (through simulated annealing) to find arrays with good autocorrelation and cross-correlation.[8] The problem is theoretically difficult and these methods are haphazard. By contrast, researchers have extensively studied sequences with good autocorrelation and cross-correlation properties[9,10] and have found applications in radar and spread-spectrum communications. We've used these sequences to generate arrays and volumes. An array is formed from a collection of various cyclical shifts of a suitable sequence, which form columns (rows) of the construction. Although it's possible to involve more than one sequence to form such an array, our constructions are based on one sequence only. When the sequence length is a prime p and the number of shifts is also p, several orderly arrangements of column shifts yield arrays with constrained autocorrelation and cross-correlation.

- *Exponential*. The shifts are powers of a primitive root of Z_p. This ensures good autocorrelation, but so far the families of such arrays with good cross-correlation are small.
- *Logarithmic*. The shifts are based on the number-theoretic index function (discrete logarithm) based on a primitive root of Z_p. This is similar to the generalized Legendre sequence defined over roots of unity, but unlike the former, the family of such arrays with good cross-correlation is small. The construction is cryptographically strong because the inversion of the discrete logarithm is a difficult problem.
- *Polynomial*. This construction assigns cyclical shifts based on polynomials of degree n with coefficients chosen from Z_p. For typical images, p can be several hundred. The choice of coefficients is a combinatorial exercise. Such a construction yields huge families of arrays with good autocorrelation and cross-correlation.

We've therefore investigated $p \times p$ arrays formed by the polynomial shift algorithm.

The constraint of prime length and square aspect are restrictive, so we're continuing to search for other arrays. Composite lengths are possible for $l = p^m - 1$, where a logarithmic shift assignment (Zech logarithm) results in arrays with good autocorrelation. The size of the families with good cross-correlation is unknown, but we expect that it's small. We can modify the square format slightly by puncturing or appending columns.

Researchers have studied sequences over prime lengths, and many of them are available. Typically, they are constructed over roots of unity—in special cases, binary. We've used maximal-length, Legendre, and generalized-chirp sequences. We can use roots of unity to construct watermarks for color and gray-scale images.[5,6]

Polynomial construction array properties

Here we present the properties of arrays constructed by using a shift polynomial of the form

$$\varphi(x) = a_n x^n + a_{n-1} x^{n-1} + \ldots + a_0$$

where all $a \in Z_p$. $\varphi(x)$ is the cyclical shift (expressed modulo p) of the column whose index (label) is x. For illustration purposes, consider p to be 509, compatible with a 512×512 image size. Consider $n = 50$.

Autocorrelation. The autocorrelation is the list of a matrix's dot products with its (cyclically) shifted versions. A vertical shift (in the column direction) of δ results in an addition of δ to a_0. A horizontal shift by ε results in a transformation of x to $x + \varepsilon$ in the shift equation. We describe the shifted matrix with the modified shift polynomial: $\varphi'(x) = a_n(x+\varepsilon)^n + a_{n-1}(x+\varepsilon)^{n-1} + \ldots + a_0 + \delta$.

The number of matching columns—that is, the columns with the same shift in identical locations in the shifted and unshifted matrices—determines the array's autocorrelation. These columns contribute p to the correlation, but all other columns contribute –1 (for standard pseudonoise columns). The number of solutions to $\varphi'(x) = \varphi(x) = 0$ determines the number of matching columns. This corresponds to the number of roots of a polynomial whose degree is at most $n - 1$. Thus, the number of such solutions is at most $n - 1$, so that the off-peak autocorrelation numbers are all bounded above by $R_{AA'} = (n - 1)p - [p - (n - 1)] \approx (n - 2)p$.

Therefore, an upper bound on the off-peak autocorrelation value normalized to the peak of p^2 is $(n - 2)/p$, which is less than 10 percent for this example. The actual correlation values range from $-p$ (no matching columns) to the maximum, in increments of $p + 1$, but the number of occurrences is constrained by the matrix balance (the sum of all entries). The shift polynomial determines each correlation value's location (a function of shift parameters δ and ε).

An example of small arrays illustrates the construction method in Figure 2 and the correlation properties in Figure 3.

Cross-correlation. Consider the two matrices A and B, which aren't shifts of each other, generated by two different shift polynomials: $\varphi(x)$ and $\psi(x)$. As in our previous example, the number of solutions to $\varphi(x) - \psi'(x) = 0$ determines the number of matching columns. This corresponds to the number of roots to a polynomial of at most n degrees. Thus, the number of such solutions is bounded above by n, so that an upper bound on the cross-correlation is $C_{AB'} = np - [p - n] \approx (n - 1)p$. This is also less than 10 percent of the peak autocorrelation.

Family size. The number of polynomials of degree n is $(p - 1)p^n$. Each matrix is available in p^2 shifts. Hence, the total number of nonequivalent matrices, generated by polynomials of degree n, is $(p - 1)p^{n-2}$. For this example, the number of nonequivalent matrices exceeds 4×10^{132}, far in excess of the number of all images and video likely to be produced.

Information capacity. Each $p \times p$ matrix can be present in p^2 shifts. Therefore, we can store $2\log_2(p)$ binary bits of information in such an array. This is almost 18 bits for this example. Where we require more information capacity, we can add watermarks together and still recover them.[11] This results in degraded autocorrelation and cross-correlation and greater image modification. The statistics appear to be Gaussian—that is, watermarks add like noise. For a sum of q watermarks, the information capacity is multiplied by q, and the standard deviation in the correlations σ is multiplied by $q^{1/2}$.

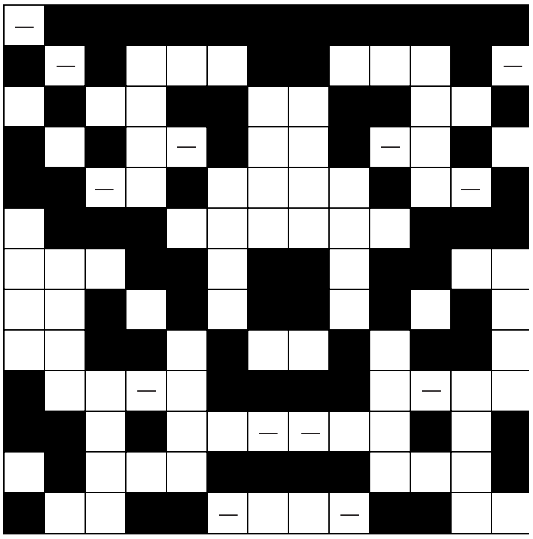

Figure 2. A 13 × 13 quadratic shift array.

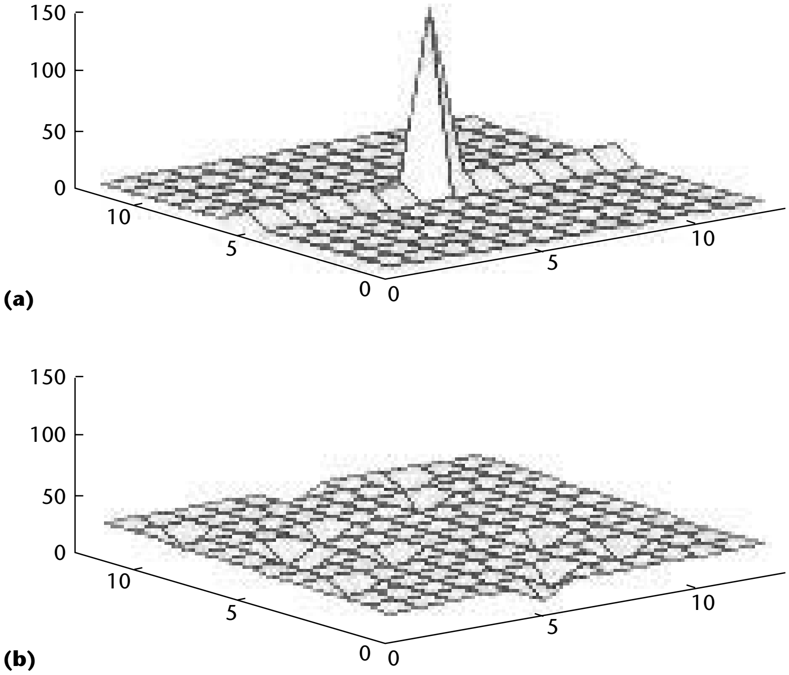

Figure 3. (a) Autocorrelation and (b) cross-correlation of a 13 × 13 quadratic shift array.

Balance. The arrays' good correlation properties depend on the off-peak sequence correlation being low. Therefore, the seed column must be near balanced (Legendre, FZC, maximal length sequence) and so are the arrays. Hence, when used as a watermark, our arrays have a minimal impact on the image mean.

Robustness against cryptographic attack. The image's low contrast regions are the most vulnerable to cryptographic attacks because portions of the array might be available to the attacker. Our arrays require accurate knowledge of the shifts of at least n columns (out of a total p) to determine the shift polynomial. Should this be insufficient protection, adding several watermarks offers cryp-

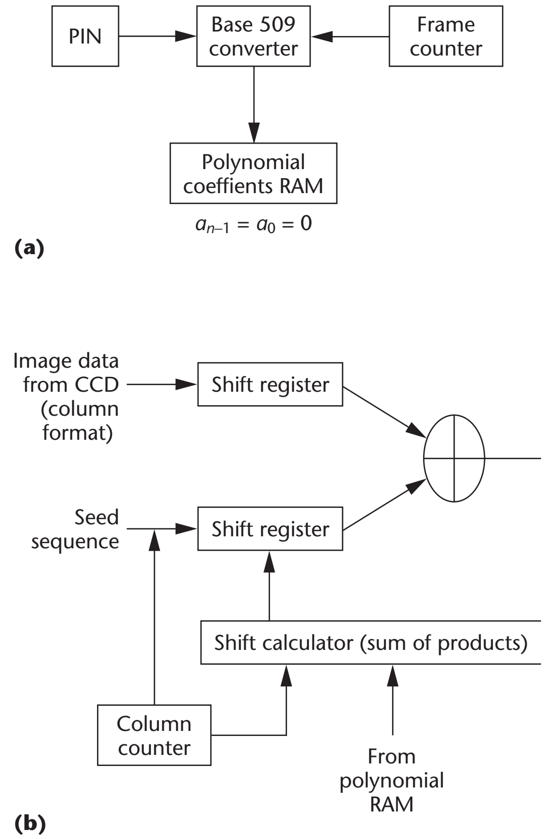

Figure 4. (a) Polynomial coefficients calculator providing a unique signature for every image and (b) an overall watermark calculator.

tographic confusion. Using cryptographically strong columns (such as Gordon Mills Welch [GMW] sequences) can enhance column security.

Implementation

We envision the arrays we just described being implemented in hardware at the analog–digital conversion stage for both digital still and video cameras.

Watermark construction

For digital still cameras, the array construction could proceed as we illustrated in the last section. Video cameras require 3D arrays, with time (frame index) forming the third dimension. We can construct $p \times p \times p$ arrays by employing polynomial shifts in the column and frame indices with a product of polynomials in the two variables—a direct extension of the theory we just outlined. We're continuing to investigate shift expressions involving polynomials in multiple variables.

Physically constructing the watermark requires computation of the shift variable $\varphi(x)$ (using modulo p arithmetic) and the cyclic rotation of a stored sequence.

Because the potential number of matrices provided by this construction is inexhaustible, systematic assignment of shift polynomials that result in distinct matrices isn't trivial. A scheme that enables assignment of a 10-digit (decimal) personal identification number (PIN), a 10-digit camera serial number, and individual image assignments exceeding 10^{30} is possible.

Camera interface

The column-based construction of our matrices is commensurate with a CCD readout. This limits the watermark to the spatial domain. We could perform the embedding with existing digital signal processing or microprocessor hardware or an EPLD. The memory requirement is minimal: n shift polynomial coefficients and the parent sequence (p terms). For shift register sequences, we only require the register coefficients, reducing memory space to a total of order $n + \log_2(p)$.

A card reader must be installed in the camera, unless it uses a smart card. In that case, we could perform the watermark generation within the card, whose output could be an analog voltage. Such external implementation requires camera clock, row–column synchronization, and other communication between the card and camera. Figure 4a shows how we can interface a PIN and frame counter to produce a unique set of coefficients used to specify the array shift polynomial. We store these coefficients in the RAM and update them as required.

We use these coefficients to calculate the sequence of rotations of each column to generate the watermark array. We can perform this calculation in an EPLD because it involves multiplication and addition modulo prime. Figure 4b schematically shows this as the shift calculator. The calculator's output controls the column sequence's shift, which a shift register or read out of RAM can generate. We add this shifted sequence to the output of a line scan from the CCD camera. This operation must be performed synchronously. (We don't show the synchronization circuitry.)

Computer interface

We designed the watermark we describe here to protect the image between acquisition and distribution. It may be incompatible with or interfere with the watermark(s) involved at the distribution stage. Therefore, it may be desirable to combine a new watermark's insertion with the removal of the previous one. This is likely to be essential where lossy compression is involved. We can successively apply such compression, and the results might not be commutative (for example, DCT-based JPEGs and fractal compression). As a result, watermarks designed to cope with such issues should be involved.

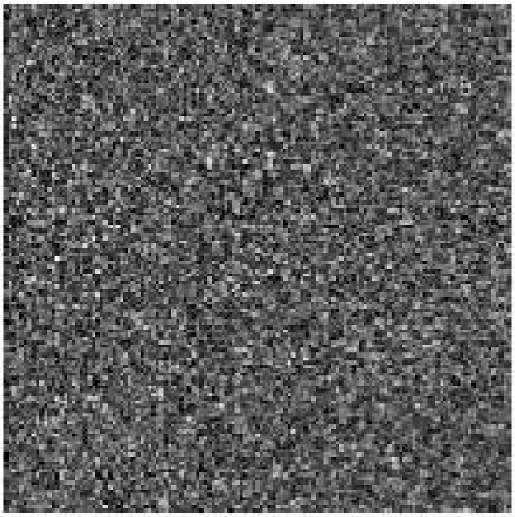

Figure 5. Superposition of four 127 × 127 quadratic shift watermarks.

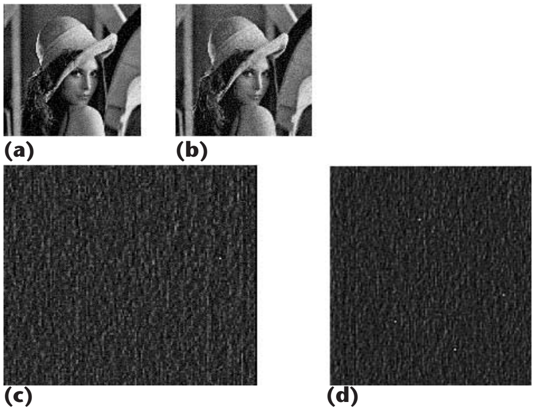

Figure 6. Four watermarks embedded in a standard image. (a) The original is 127 × 127 × 8 bits. (b) We added 127 × 127 binary watermarks in the spatial domain for four different m values. (c) The extraction of one out of the four watermarks. (d) The superposition of all four extracted watermarks.

Multiple arrays

The number of cyclic shifts available restricts the arrays' information capacity. This is equivalent to about 16 bits in the earlier example. Superposition of multiple watermarks on the same image enhances this capacity. Figure 5 shows the superposition of four quadratic shift arrays on a null image.

Figure 6 shows the embedding in the spatial domain of Figure 5's watermark in a standard image and the recovery of each of the watermarks (each had a deliberately unique cyclic shift). We took $p = 127 = 2^7 - 1$ and chose one of the 18 maximal length sequences (m-sequences) of length 127, as a seed column for the construction. For each of $m = 1, 2, \ldots, 126$, as the value of the multiplier, we considered the 127 × 127 array constructed from the seed column and with a sequence of relative shifts for the subsequent columns: $0, m, 2m, \ldots, 126m \pmod{127}$. This is equivalent to a quadratic shift polynomial of the form

$$\varphi(x) = \frac{1}{2} m\left(x^2 + x\right)$$

For the autocorrelation, $\varphi(x) = (x + \varepsilon) + \delta$ has at most one solution for x, corresponding to zero or one columns matching, apart from the peak at $\delta = \varepsilon = 0$ (where all the columns match). Thus, the normalized off-peak autocorrelation values are

$$\frac{-1}{p}, \frac{1}{p^2}$$

The cross-correlation between different arrays takes on the normalized values of

$$\frac{-1}{p}, \frac{1}{p^2}, \frac{p+1}{p^2}$$

Therefore, all off-peak correlations are at most of order $1/p$. This is less than 1 percent for the 127 × 127 image. (Hall, Osborne, and Tirkel[12] give a detailed analysis of this special case.)

We selected 12 of these matrices with a random number generator and superimposed all 12 on the Lena image (see Figure 6), on the same 127 × 127 pixels. Each of these matrices was cyclically shifted in two dimensions. We did this with many choices of the set of 12 arrays. In all cases, we accomplished unambiguous retrieval of each of the 12 watermarks by using each array as a template without recourse to the unmarked image. The process automatically retrieves the horizontal and vertical shifts of each of the 12 matrices, which lets us store 168 (=12 × 14) bits of information in the composite watermark. Retrieving the individual arrays was possible because of the low cross-correlation values for each pair of the 126 available matrices and each matrix's low off-peak autocorrelation values.

The penalty for increasing the watermark's capacity is an increase in errors: false or missed detection and recovery. To understand the errors involved in the recovery of multiple watermarks, we first analyze the errors involved in recovering a single watermark.

Errors in watermark recovery

In watermarking, we can consider the image the dominant source of interference with the recovery process. Typically, we can assume the image is uncorrelated with the pseudonoise arrays and sequences used to carry information. Therefore, in the correlation process, image components add incoherently, but the autocorrelation process is coherent. Hence, the watermark output rises with the number of entries n, and the cross-correlation with the image rises as $\sqrt{n}$. For a full correlation of a $p \times p$ array, the processing gain is $20\log(p)$ dB. In this respect, the situation is similar to that in the communication application of the spread spectrum. This is a simplification because image statistics can cause significant departures from this average result. For example, for binary watermarks, skews in the image distribution can result in a correlation bias. Such a skew can occur naturally or as a result of histogram manipulation in image editing.

We can divide the interference from other arrays embedded in the image into two classes: synchronous and asynchronous. The former occurs when we embed multiple arrays with low cross-correlation to increase the watermark's information capacity. This cross-correlation is like a baseline offset on the postcorrelation output. It adds arithmetically to the autocorrelation. Because the shifts (ε, δ) carry the information and this background is unpredictable, we can calculate the cross-correlation's average effect from the variance or standard deviation. These are low for all the constructions we discuss here. However, the distribution of such correlations is skewed and has a long positive tail. Thus, the probability of missed detection is low and controllable, but the probability of false alarm depends on the maximum positive cross-correlation value. This shows the necessity of designing arrays with constrained maximum cross-correlation.

Error statistics

We consider two cases of watermark recovery, where the watermark is based on the arrays constructed from polynomial-based shift sequences. Frist, we analyze a single array embedded in an image, which is the only source of interference. When multiple arrays are embedded, we must consider the cross-correlation between such arrays.

Single array. The probability of false detection of a single watermark in an array of size p^2 is

$$p_{\mathrm{FD}} = 1 - (1 - p_e)^{p^2}$$

where p_e is the probability that a random variable with variance σ exceeds an error threshold T_e.

The probability of missed detection of the watermark is $p_{\mathrm{MD}} = p_d$, where p_d is the probability that the same random variable doesn't exceed $p^2 - T_d$, with p^2 being the autocorrelation peak. Such an event would reduce the autocorrelation peak below the threshold. For a random process with Gaussian statistics, the probability that such a random variable exceeds a threshold T_e is

$$p_e = \frac{1}{2}\left[1 - erf\left(\frac{T_e}{\sqrt{2}\sigma}\right)\right]$$

Strictly speaking, we must include the effects of the off-peak autocorrelation when analyzing false detection. In the case of multiple embedded arrays, we should include the cross-correlation between the desired array and all the others in the probability of missed detection. Both of these effects are small for the image size considered and for the array construction with low-degree polynomials. These correlation effects are deterministic, but because the arrays' shifts are unknown, their effects can only be treated statistically. In most applications, it's desirable to make these two error events equally probable and sufficiently small. When these probabilities are less than 10^{-10} (comparable with error rates for digital modems), the postcorrelation PSNR must exceed 23.14 dB, with T_e set at $0.557p^2$. For a 509 × 509 array, the processing gain is 54.13 dB. Therefore, the precorrelation PSNR must exceed –31 dB. This is 9 dB above the recommended level for image distortion (–40 dB). Should such an array be embedded at the recommended strength, the probabilities of false or missed detection rise to 0.34. Embedding at the recommended level is consistent with acceptable error rates for arrays exceeding 1,500 × 1,500 entries. This applies to all watermarking schemes relying on correlative recovery.

Multiple arrays. We can achieve a different scenario by using a combination of multiple arrays and an error-correcting code. For the 509 × 509 array in the last section, consider embedding the information-carrying array plus seven parity arrays of a binary Bose Chauhuri Hocquenghem, or BCH, (127, 22, 23) code. The information in each array is contained in its cyclical shift, which equates to almost 18 binary bits. Eight arrays carry almost 144 bits, sufficient for the above BCH code (127 bits).

For an undetected or uncorrected 10^{-10} error rate, the postcorrelation PSNR must exceed 16.9 dB, with $T_e = 0.73p^2$. The watermark PSNR referenced to the image power is –37.2 dB. Because we added the eight arrays incoherently, as power, the total image distortion is –28.2 dB. Thus, the scheme costs 2.8 dB of distortion, without affecting the error rates. We could recover at least 3 dB by using the 4 bits extra capacity of the compound watermark for trellis or turbo coding.

The advantage of such a compound watermark is enhanced security and less obtrusiveness, particularly in low-contrast regions of the image. The construction is difficult to invert, even in the absence of an image. Consider a cryptanalyst who has obtained the compound watermark, either by successfully subtracting the original image or through a security lapse. The analyst can deduce the nature of the original column used in the construction from Fourier analysis and correlation with known sequences of appropriate length. However, each constituent array's shift sequence is difficult to deduce because each column of the compound watermark is a superposition of eight columns. **MM**

Acknowledgment

We thank Ron van Schyndel for performing the experiment in Figure 6. This article is based on a paper we presented at ACM 2000.[13]

References

1. A.Z. Tirkel et al., "Electronic Watermark," *Proc. Digital Image Computering Techniques and Applications* (DICTA-93), Macquarie Univ., Sydney, 1993, pp. 666-673.
2. F. Hartung and M. Kutter, "Multimedia Watermarking Techniques," *Proc. IEEE*, IEEE Press, Piscataway, N.J., vol. 87, no. 7, July 1999, pp. 1079-1107.
3. F.A.P. Petticolas, R.J. Anderson, and M.G. Kuhn, "Information Hiding—A Survey," *Proc. IEEE*, vol. 87, no. 7, July 1999, pp. 1063-1077.
4. F.J. MacWilliams and N.J.A. Sloane, "Pseudo-random Sequences and Arrays,*" Proc. IEEE*, vol. 64, IEEE Press, Piscataway, N.J., 1976, pp. 1715-1729.
5. R. van Schyndel, A.Z. Tirkel, and I.D. Svalbe, "A Multiplicative Color Watermark," *Proc. IEEE-EURASIP Workshop Non-Linear Signal and Imaging Processing*, Bilkent Univ. Ankara, Turkey, 1999, pp. 336-340.
6. I.D. Svalbe, A.Z. Tirkel, and R.G. van Schyndel, "Discrete Angle Watermark Encoding and Recovery," *Proc. Int'l Conf. Pattern Recognition*, IEEE CS Press, Los Alamitos, Calif., 2000, pp. 246-249.
7. K. Ohue, "A Generation Method of Periodic Orthogonal Numerical Sequences with Small Maximum Amplitude for any Period," *IEICE Trans. Fundamentals*, E80-A, no. 10, 1997, pp. 2016-2021.
8. J. Bernasconi, "Low Autocorrelation Binary Sequences: Statistical Mechanics and Configuration Space Analysis," *J. de Physique*, vol. 48, no. 4, Apr. 1987, pp. 559-567.
9. D. Everett, "Periodic Digital Sequences with Pseudonoise Properties," *G.E.C. J.*, vol. 33, no.3, 1966, pp. 115-126.
10. M.R. Schroeder, *Number Theory in Science and Communication*, third ed., Springer, New York, 1997.
11. R. van Schyndel et al., "Algebraic Construction of a New Class of Quasi-Orthogonal Arrays in Steganography," *Proc. SPIE Electronic Imaging 1999*, SPIE Press, Bellingham, Wash., pp. 354-364.
12. T.H. Hall, C.F. Osborne, and A.Z. Tirkel, "Families of Matrices with Good Auto and Cross-Correlation," to be published in *Ars Combinatoria*, 2002.
13. A.Z. Tirkel and T.E. Hall. "Watermarking at Point of Origin," *Multimedia and Security Workshop at the 7th ACM Int'l Multimedia Conf.*, GMD, Darmstadt, 2000, pp. 135-138.

Andrew Z. Tirkel is a director of Scientific Technology and a visiting fellow in the Department of Mathematics and Statistics at Monash University. His interests include sensors, radar, and signal processing. He developed the spread-spectrum watermark with Charles Osborne. He has a BSc and PhD in physics from Monash University. He is an IEEE senior member.

Thomas E. Hall is a reader in mathematics at Monash University. His interests are in algebraic theory of semigroups and combinatorics for communications. He has a BSc and PhD in mathematics from Monash University. He is an Australian Mathematical Society honorary life member and an Australian Mathematical Society fellow.

Readers may contact Tirkel at the Dept. of Mathematics and Statistics, Monash Univ., PO Box 28M, Victoria 3800, Australia, email atirkel@bigpond.net.au.