EURASIP Journal on Advances
in Signal Processing

RESEARCH Open Access

Extended families of 2D arrays with near optimal auto and low cross-correlation

I. D. Svalbe[1*] and A. Z. Tirkel[2]

Abstract

Families of 2D arrays can be constructed where each array has perfect autocorrelation, and the cross-correlation between any pair of family members is optimally low. We exploit equivalent Hadamard matrices to construct many families of p $p \times p$ arrays, where p is any *4k-1* prime. From these families, we assemble extended families of arrays with members that exhibit perfect autocorrelation and next-to-optimally low cross-correlation. Pseudo-Hadamard matrices are used to construct extended families using $p = 4k + 1$ primes. An optimal family of *31* 31×31 perfect arrays can provide copyright protection to uniquely stamp a robust, low-visibility watermark within every frame of each second of high-definition, *30* fps video. The extended families permit the embedding of many more perfect watermarks that have next-to-minimal cross-correlations.

Keywords: Pseudo-noise arrays, Correlation, Discrete projection, Finite Radon transform, Digital watermarks

1 Introduction

A digital watermark is a kind of marker covertly embedded in a noise-tolerant signal such as an audio, video or image data. It is typically used to identify ownership of the copyright of such signal. Images are 2D, so 2D watermarks are required. Video can be considered as sequences of images. Two-dimensional perfect arrays are fundamental in coded aperture imaging, e.g. phased array antennas, arrays of sound sources or flashing tomography. Further, arrays are used in higher-dimensional signal processing applications such as time-frequency-coding, spatial correlation or map matching, parallel processing in one-dimensional correlation, arrays for built-in tests of VLSI-circuits or 2D measuring techniques, e.g. optical systems [1]. They are also useful in image and watermark registration [2]. Families of 2D arrays with low off-peak autocorrelation and low cross-correlation are useful in digital watermarking to identify the owner of an image, or to embed multiple arrays in an image to increase the amount of hidden information in the watermark.

This paper presents methods to construct large families of discrete 2D arrays over the integers, or roots of unity. This is unusual. Among sparse arrays, Costas arrays have optimal autocorrelation, but there are no large families with optimal, or even good cross-correlation [3]. Among dense arrays, m-sequences have perfect autocorrelation, but only small families, called maximal connected sets have good cross-correlation [4]. The full set of m-sequences may have very high cross-correlations, particularly for composite lengths [5]. This paper builds on prior work [6] that built optimal families of p $p \times p$ arrays, where $p = 4k\text{-}1$ is prime. Those 2D families are optimal because each array within a family has a two-valued periodic autocorrelation, and all of the cross-correlations between any pair of family members are the lowest possible. New methods are described here to find disjoint families of these optimal 2D arrays and then construct extended families of $p \times p$ arrays that exhibit perfect autocorrelation and cross-correlations that are asymptotically optimal. This means a large fraction of all cross-correlations between family members are the lowest possible. The remaining fraction of cross-correlation values are distributed across a narrow range of next-to-lowest possible values. This work relies on two theoretical concepts. The first key is that the process of discrete projection preserves the correlation properties of any array [7, 8]. This means the correlation properties of any nD array are inherited through the correlation properties of the 1D projections of that nD

[*] Correspondence: imants.svalbe@monash.edu
[1] School of Physics and Astronomy, Monash University, Clayton 3800, Australia
Full list of author information is available at the end of the article

array. The second key is the remarkable structure of Hadamard matrices [9], which specify how to tile a list of M elements in M ways so that all M tilings are orthogonal i.e. the sum of their dot products is 0. The M tilings are said to be maximally different. We show that a family of arrays with low cross-correlation between matching array projections must be maximally different in the Hadamard sense.

To achieve perfection in an array requires precise synchronization between all array elements. Perfect arrays have the additional useful property that their structure, superficially, appears to be random. One application for the new arrays constructed here is to transparently embed a family of 2D planes in frames of video or any digital media. Such arrays serve as secure, robust and low-visibility spatial or transform domain spread-spectrum watermarks [10]. The cross-correlation between watermarking arrays embedded in different frames is the lowest possible. This results in the lowest false positives. In addition, the off-peak autocorrelation of each watermark array is −1, resulting in the lowest missed detection rate, and allows the watermark to be used as a registration mark [2]. The number of frames with unique marks is variable, up to *p.* Each mark can be repeated as needed. Additionally, because of their low cross-correlation, multiple arrays can be embedded in the same frame, to increase the data payload.

The array family size is bounded by p to maintain the optimal correlation performance. However, many more such families can be generated using different Hadamard matrices of size $p + 1$. These can be obtained by negating rows or columns or by interchanging rows or columns. Such Hadamard matrices are called equivalent. There are billions of them for practical values of *p.* In addition, there are inequivalent matrices. For $p = 31$ (the largest value for which this is known), there are 13,710,027 inequivalent Hadamard matrices [11]. Consequently, the number of families is virtually inexhaustible. An algorithm for finding families with lowest mutual cross-correlation is being investigated.

Our scheme is suited to watermarking to meet the proposed Society of Motion Picture & Television Engineers (SMPTE) watermarking standard, where high and flexible data payload and short, variable watermark duration is required

> *The binding method must accommodate a payload of a minimum of 25 bytes (200 bits) to carry simultaneously Ad-ID (the advertising industry standard unique identifier for all commercial assets airing in America) and EIDR (the Entertainment Identifier Registry, is a global unique identifier system for a broad array of audio visual objects, including motion pictures, television, and radio programs) (each of which is 96 bits long), along with indicator(s) (such as enumerated values) to label each ID, plus overhead. The binding mechanism shall allow the transition to or from a uniquely identified or unidentified piece of content to be detected within 1 second of the transition. Transition detection does not necessarily require the recovery of the content ID* [12].

A previous array-based watermarking method developed by one of the authors [13] uses 3D arrays to watermark the video. It cannot meet the SMPTE requirements, because the array is long in the time dimension and requires 3–4 min per watermark. Also, it struggles to meet the data payload, and the construction is not flexible, as required.

A second application is to encrypt information within any digital data by encoding multiple arrays within a local region of the data. The encrypted message content is decoded by identifying the presence of each individual array and specifying its exact spatial location. Extended families of arrays provide unique owner identification for more users, or encrypt a more diverse information payload. It is vital for both applications that all embedded arrays can be recovered and uniquely identified. Successful implementation requires arrays that appear to be random but simultaneously have a perfectly sharp autocorrelation and the lowest possible cross-correlation with all other arrays.

The paper is organized as follows: Section 2 presents the mathematical background on the construction and correlation properties of families of arrays suitable for watermarking. It also introduces maximal and minimal correlations between 1D sequences and nD arrays and reviews the link between nD arrays and their discrete 1D projections using the finite Radon transform. Section 3 presents new results for 2D $p \times p$ arrays to extend the size of disjoint families that retain perfect autocorrelation and near optimal cross-correlation. Hadamard matrices are used to make perfect arrays that are maximally different. Section 4 deals with the embedding and extraction of watermarks in images and video. Section 5 discusses the experimental method used that presents results and discusses embedding applications for optimal arrays.

2 Algebraic arrays for watermarking

Digital watermarking using spread-spectrum techniques started in 1992, with the first publication appearing in 1993 [14]. This used m-sequences embedded in images line by line. Considerable interest was generated by this, mainly centered on the methods of embedding and extraction, resistance to compression, compatibility with Human Visual System, resistance to incidental distortions and deliberate attacks and diversification to other media, such as audio and video. Our group focused on ‘what to embed’. It was clear that 2D arrays were needed. The requirements were:

1 *Sizes commensurate with image format*
2 *High peak autocorrelation and low off-peak autocorrelation to reduce missed detection and false alarm rate respectively*
3 *Large and flexible family size, with all cross-correlations being low*

At the time, the only known 2D arrays that partially satisfied these criteria were the small and large Kasami sets of sequences folded into 2D. Our group then focused on Costas arrays, invented by John Costas to resolve radar ambiguities, and independently by Edgar Gilbert. These were unsuitable, because of their sparsity, resulting in low peak autocorrelation. Also, families with low cross-correlation were tiny. It took one of the authors 20 years, and collaboration with T.E. Hall and the late O. Moreno to overcome the sparsity issue and the small family size. The sparsity was addressed by using the method of composition. A Costas array or related pattern of dots, with one dot per column was used as a shift array to compose a 2D array whose columns were shifts of a binary sequence with ideal autocorrelation. This method of composition and column substitution was then generalized by finding shift arrays which produced large families, with optimal cross-correlation. The method was extended further, to higher dimensions, and in fact, 3D arrays were successfully embedded in and extracted from video and survived H264 and H265 compression [13]. A limitation of such arrays is their size $p \times p \times (p^2\text{-}1)$. The $p^2\text{-}1$ is difficult to adapt to video watermark requirements, as discussed in Section 3. Also, while roots of unity alphabets are commensurate with finite field algebra, apart from the binary case, they are not readily accommodated in image data. It is possible to transform image data to accept higher roots of unity, as in [15], but such transformations are cumbersome. By contrast, greyscale arrays are naturally suitable for embedding in images.

In this paper, we introduce a different array construction which overcomes the above limitations and provides $p \times p$ arrays for watermarking of individual video frames. p such arrays exhibit optimal cross-correlation. This permits the embedding of more arrays in a single frame to increase data payload or to satisfy the watermark duration requirements. The method we use in this paper is different from the method of composition and column substitution. It relies on projective geometry. We partition a $p \times p$ array into $p + 1$ projections and assign +1 or −1 values to these projections according to rows of a commensurate Hadamard matrix. Such arrays are ideally suited to image and video watermarking, and we show how they can be adapted to greyscale and extended to larger families.

2.1 Definition of correlations

Our concern here is correlations between discrete arrays in n-Dimensions (nD). We define arrays $A(k, l)$ and $B(k, l)$ to be 2D discrete arrays, with integer indices $0 \le k,\ l < N$. The value at input array location $(k,\ l)$ is usually a signed integer, or a root of unity. The periodic cross-correlation (CC_p) between arrays A and B, measured at location $(r,\ s)$ is defined as:

$$CC_p(r,s) = A \otimes B = \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} A(k,l) \cdot B(k+r, l+s) \quad (1)$$
$$= F^{-1}[A(u,\ \ v) \cdot B(u,v)]$$

where $A(u,v) = F[A(k,l)]$, $B(u,v) = F[B(k,l)]$, and F, F^{-1} denotes forward and inverse discrete Fourier transform (DFT), respectively, and '·' denotes the element-by-element dot product. The cross-correlation $CC_p(r,s)$ is an array with $N \times N$ values under periodic boundary conditions. For aperiodic (zero-padded) boundary conditions, the array $C_a(r,s)$ has size $(2\ N\text{–}1) \times (2\ N\text{–}1)$. When $B = A$, $AC_p = A \otimes A$ denotes periodic autocorrelation (AC), AC_a denotes aperiodic autocorrelation.

2.2 Review of perfect autocorrelations

A 1D sequence of length N that has a periodic autocorrelation with a single peak with value N and all $N\text{-}1$ off-peak values being 0 is a perfect sequence. Such sequences are unbalanced, and fail most randomness criteria such as Golomb's postulates [16]. The values of a pseudo-noise (pn) sequence are distributed about 0, preferably symmetrically, and have random, noise-like structure. A pn sequence of length N is comprised of signed integers, roots of unity or other alphabets. The mean of these sequences is 0 for even length or ± 1 for odd length. The periodic autocorrelation of perfect or pn sequences for integer shifts is a delta function. Equivalently, perfect and pseudo-noise sequences are spectrally flat.

For higher dimensions, we use arrays instead of sequences. Perfect arrays exist for selected sizes, but remain unbalanced. The periodic autocorrelation of perfect and pn arrays is an nD δ function. Perfect and pn arrays are spectrally flat in nD. Here, we restrict the arrays to be $p \times p$ planes in 2D, with integer values, usually $0,\ \pm 1$. The merit factor (MF) is one measure of the quality of correlations. It is defined as the ratio of the squared peak correlation value to the sum of all squared off-peak correlation values. The merit factor can be periodic (MF_p) or aperiodic (MF_a). A second metric for correlations is the periodic correlation ratio R_p (and aperiodic ratio R_a) defined as the peak value divided by the next largest correlation value. R is often called the 'peak-to-side-lobe' ratio. For perfect arrays, $MF_p = \infty$, $R_p = \infty$. For pn hypercube arrays of size N in nD, the maximum periodic values are $MF_p = N^n\text{-}1$ and $R_p = N^n\text{-}1$. Non-periodic data must deal with aperiodic correlations. Generally, the aperiodic MF_a and R_a values are significantly lower than MF_p and R_p values for the periodic case.

The above measures relate to the array itself. When an array is embedded in noise-tolerant data, the cross-correlation of the array with the marked data exhibits a peak when the reference array is aligned with the embedded array. The off-peak values are typically Gaussian distributed, as discussed in Section 4, and shown in Fig. 1a. Figure 1b shows the time sequence of the correlations.

The distribution is characterized by standard deviation σ. The peak to σ ratio (signal to noise ratio or SNR) determines the probabilities of missed or false detection, based on the cumulative normal distribution. The SNR indicates how easy it will be to recover an array that has been embedded in digital data, where the host data can be regarded as noise. The SNR scales with both the size N and the dimension (n) of the embedded signal.

Arrays chosen for embedding should be spectrally flat while having apparently random structure to minimize their perceptibility. They should utilize a limited alphabet, preferably with using $+1$ or -1, except for an occasional 0. Such arrays are easy to embed and are highly efficient.

2.3 Review of cross-correlations

Tirkel et al. [6] gives constructions of pn arrays of size $p \times p$, where $p = 4k\text{-}1$ is prime. The elements are a single 0 and equal numbers of ± 1. The autocorrelation MF_p is $p^2\text{-}1$. The more challenging aspect is to constrain the cross-correlations between pairs of arrays to be optimally low and to then construct a family of arrays with as many members as possible. The lowest possible periodic cross-correlation values between a family of p of these $p \times p$ arrays in 2D can be shown [6] to be 0 at zero shift, and either $+p$ or $-p$ at all other shifts. Correlation bounds for families of sequences have been studied for an arbitrary alphabet by Welch [17] and for roots of unity by Sidelnikov [18]. These concepts can be extended to arrays. The peak correlation of the family of p $p \times p$ arrays in [6] is consistent with the Welch bound [17], making these families optimal.

A conjecture of Golay [19], that the highest aperiodic merit factor for sequences is achieved by Legendre sequences of length p at shift $\frac{p}{4}$, is now extended to two dimensions, where the maximum $\mathrm{MF_a}$ is obtained for Legendre $p \times p$ arrays occurs at or near shift $\left(\frac{p}{4}, \frac{p}{4}\right)$.

Much of the initial work on constructing arrays used finite field theory to expand or fold 1D m-sequences sequences into higher dimensions [20]. There is clear link between folding sequences and discrete projection. We exploit the geometric insights offered by discrete projection theory to construct perfect or pn arrays in nD from their lower dimensional projected views. Discrete projection theory [21] quantizes the continuous space Radon transform used for the tomographic reconstruction of images from projected views. A short review of discrete projection is given in the next section.

2.4 Discrete projections as pn sequences

Traditionally, arrays have been analysed as concatenations of sequences using a row by row approach [22], or, for relatively prime dimensions, by a single pass diagonal [20], equivalent to the application of the Chinese Remainder Theorem. Our approach uses discrete projections. We utilize the finite Radon transform (FRT) to obtain discrete projections for nD arrays of size p, where p is prime [22]. The FRT permits any nD array to be represented exactly as a set of 1D projections, with each projection having length p. The number of 1D FRT projections of an nD array is $\sum_{i=0}^{n} p^i$. The FRT has a very simple inverse transform, the un-filtered back-projection, whereby any nD array can be reconstructed exactly from its 1D projections (see Fig. 2). Black = $+1$, red = -1 and white = 0.

We present a 2D example. FRT projection of a $p \times p$ array results in a set of $p + 1$ 1D sequences. Applying the inverse FRT to these 1D sequences re-assembles the same $p \times p$ array. The FRT, $R(t, m)$, of array $A(k, l)$, is defined (2) as the sum of the array elements that intersect p parallel digital straight lines, $k = ml + t$, where k, l, m and t are all integers and $<n>$ means the value of n modulo p. The back-projected inverse FRT (3) is similar to the forward-projected FRT (2), with re-projection done at the complemented projection angle ($p\text{-}m$), adding an extra plane from re-projecting $m = p$ and normalizing by the sum, obtained from any m, of all projected values.

$$R(t,m) = \sum_{k,l=0}^{p-1} A(ml+t,k) \tag{2}$$

$$A(k,l) = \sum_{m,t=0}^{p-1} R(-ml+t,k) + \sum_{t=0}^{p-1} R(p,t) \tag{3}$$

Here, t is the line intercept. The starting position of the line traditionally is defined along the top row of the data, $0 \le t \le p$. The slope of each line is a rational fraction, $m{:}1$ i.e. stepping m pixels across for each pixel down and ($0 \le m \le p$). The line $m = 0$ is traditionally the column direction, $m = 1$ is the diagonal 45° line, and $m = p$ means lines in the row direction.

The key concept is that discrete projection preserves correlation properties. The discrete central slice theorem [7, 8] proves that the 1D discrete projection of any 2D autocorrelation array is the same as the 1D autocorrelation

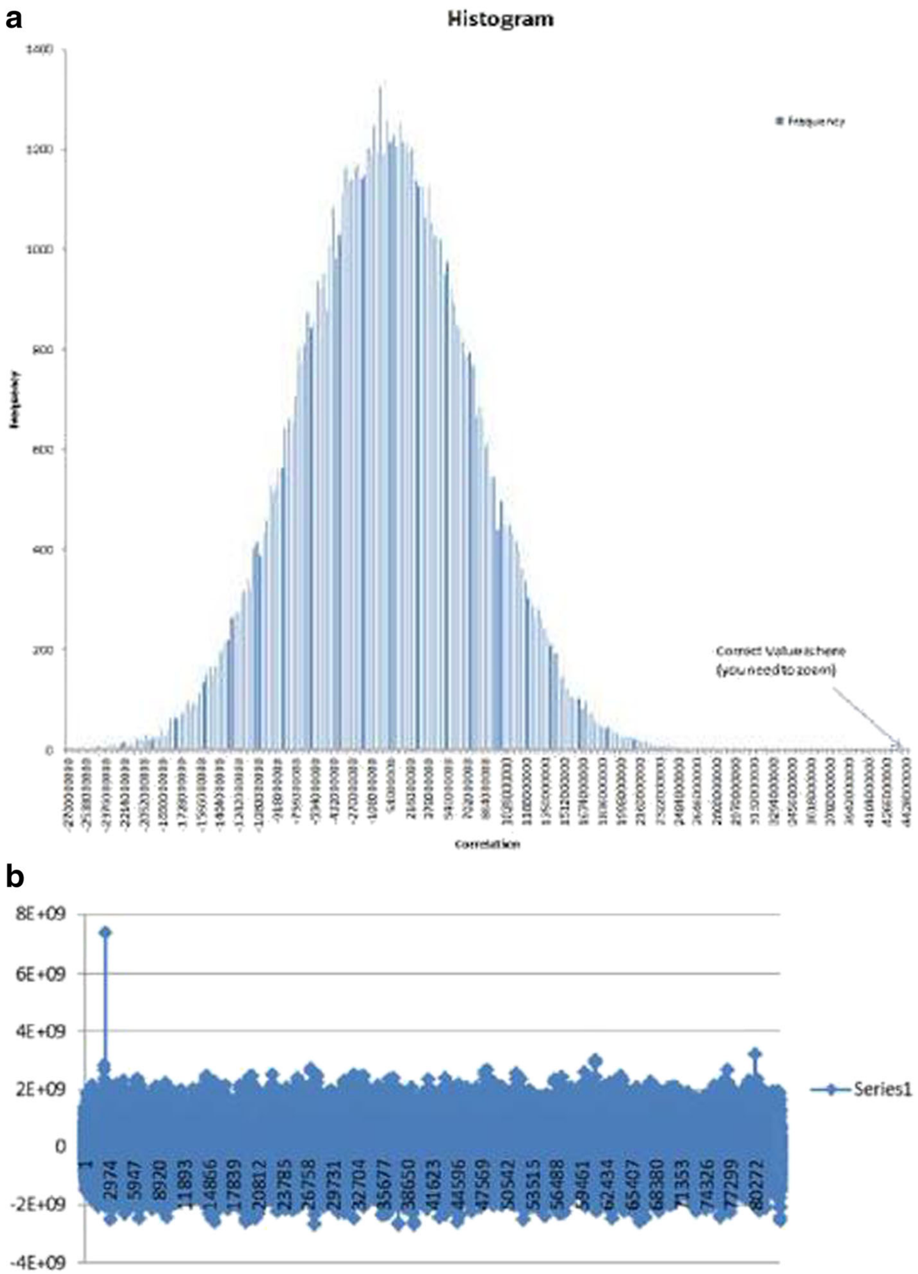


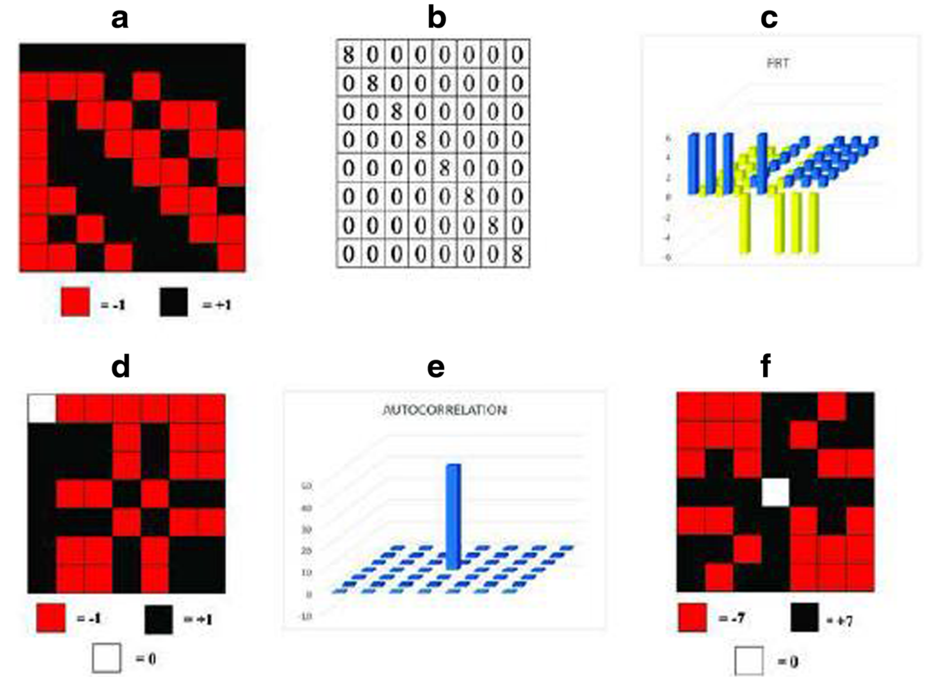
Fig. 1 **a** Histogram of correlations between template array and watermarked video. **b** Time sequence of correlation of template array with watermarked video

of the same 1D projected view of the original 2D array. Autocorrelation of discrete projections, and by analogy, cross-correlation, follows definition (1), where role of array values is replaced by the projected sums of array values.

The FRT reconstructs a unique 2D array from any fixed set of $p + 1$ 1D projections. The joint correlation properties of a set of $p + 1$ 1D projections determines the correlation properties of the 2D array that is reconstructed from those projections.

The spectral content of any 2D array can be assembled, one projection at a time, according to the spectral content of the $p + 1$ 1D projections from which the array is reconstructed. Any 1D sequence with δ function autocorrelation is spectrally flat i.e. equal magnitude at each discrete frequency. A 2D array built from $p + 1$ such projected views will also be spectrally flat and exhibit perfect 2D autocorrelation.

In the opposite direction, a low cross-correlation will result between any pair of arrays whose 1D projections,



8	0	0	0	0	0	0	0
0	8	0	0	0	0	0	0
0	0	8	0	0	0	0	0
0	0	0	8	0	0	0	0
0	0	0	0	8	0	0	0
0	0	0	0	0	8	0	0
0	0	0	0	0	0	8	0
0	0	0	0	0	0	0	8

Fig. 2 **a** A Hadamard matrix, H_8. **b** $H_8{*}H_8^T$ is diagonal: each row of H_8 is maximally different. **c** An FRT for $p = 7$. The eight rows are spectrally flat 1D projections. The seven sign patterns on rows 2–8 of (**a**) build seven distinct FRTs like (**c**). **d** The 7×7 array reconstructed from FRT (**c**). All seven arrays exhibit perfect autocorrelation (**e**). Their 21 cross-correlations are all optimally low, like (**f**)

taken along matching directions, all have low 1D cross-correlations. The discrete central slice theorem also ensures that the $p + 1$ 1D Fourier transforms of the 1D FRT projections tiles each frequency of the $p \times p$ 2D Fourier space exactly and only once. An array, its FRT and DFT are bijective mappings, see (1). If all the 1D sequences have 0 sum, then the 0 frequency term also has 0 amplitude, producing a 2D pn array.

2.5 Families of arrays with optimally low cross-correlation

A *0* cross-correlation implies a *0* dot product for the Fourier transforms of these arrays i.e. $A(u, v) \times B(u, v) = 0$. This, in turn, requires at least one of the Fourier coefficients in either array to be 0 at each frequency. This condition is inconsistent with the requirement that an array must be spectrally flat to have a perfect autocorrelation. The cross-correlation between a pair of perfect arrays or between projections of these arrays cannot be *0*. As a corollary, any array that has all 1D projected rays sum to *0* in any direction cannot have perfect autocorrelation. As a further corollary, all 1D discrete projections of an nD array must be spectrally flat for the array to be spectrally flat and have perfect autocorrelation. Any given fixed set of 1D projections can only reconstruct one, unique array: an array and its set of FRT discrete projections is bijective. To construct a different array, one or more of these projections must change. If we change just the sign of one pn projection, that 1D projection autocorrelation remains perfect, but the array that is reconstructed from this changed set of projections still has perfect autocorrelation, but now has a different spatial structure. The cross-correlation between the changed array and the original array will be strong, but less than perfect.

We can change the signs of the 1D projections of a perfect autocorrelation array A_1 to produce a perfect autocorrelation array A_2, such that $A_1 \otimes A_2$ has the lowest possible values. This means the sum of all the values that are re-projected across the 2D array from each 1D projection should be as low as possible. That means half of the $p + 1$ projections used to reconstruct array A_1 must have opposite signs to those of the $p + 1$ projections used to reconstruct array A_2. The 2D FFT of $A_1 \otimes A_2$ is partitioned by signs that match the signs of the 1D FRT projections. All of the signs of the 1D projections in A_1 need to be orthogonal (or maximally different) to all of the signs of the 1D projection for A_2. Under these conditions, $A_1 \otimes A_2$ has optimally low cross-correlation.

To construct a family of three perfect autocorrelation arrays (A_1, A_2 and A_3) with $A_1 \otimes A_2$, $A_1 \otimes A_3$ and $A_2 \otimes A_3$ each being orthogonal requires that 50% of the signs of the $p + 1$ projections assigned to build array A_1 and array A_2 must be opposite, as must be 50% of those between A_1 and A_3, as must be 50% of those between A_2 and A_3.

This brings us to invoke our second key theoretical concept: the Hadamard matrix [5]. An orthogonal assignment can be achieved, for up to p arrays, by using the non-trivial rows of an $M \times M$ Hadamard matrix, where $M = p + 1$. This is possible for $p = 4k–1$, since Hadamard matrices only exist in sizes that are multiples of 4 [9]. There are several inequivalent forms of Hadamard matrices: Sylvester, Paley, Walsh and Tonchev [9]. Williamson matrices have some unbalanced rows and are avoided.

We can construct a family of perfect autocorrelation, low cross-correlation arrays by using the non-trivial rows of an $M \times M$ Hadamard matrix as p orthogonal ways to select the pattern of sign changes for each of p sets of $M = p + 1$ 1D projections. This construction generates a family of p $p \times p$ arrays, $A = \{A_1, A_2, \ldots A_i, \ldots A_p\}$, where each array inside family A has perfect autocorrelation $A_i \otimes$ for all i, and optimally low cross-correlations $A_i \otimes A_j for\ \ i \neq j$. Figure 2 shows one member A_i from a family A of 7×7 arrays, the 7×8 1D FRT projections used to make A_i and the 8×8 Hadamard matrix used to generate the 7 FRT projection sets that make $\boldsymbol{A}$.

2.6 Families of perfect arrays

Constructing a family of p perfect autocorrelation and optimally low cross-correlation $p \times p$ arrays is possible in 2D for any $p = 4k–1$, since the projection set has size $p + 1 = 4k$. For prime $p = 4k + 1$, we can construct a balanced $p \times p$ perfect array by flipping the signs of $2k + 1$ spectrally flat 1D projections. But Hadamard matrices of size $4k + 2$ do not exist.

However, we show in Section 3 that the cross-correlation between different families of $p \times p$ arrays can be engineered to produce arrays with asymptotically low correlation.

3 Correlations between perfect array families

Section 2 reviewed how to construct families of p $p \times p$ arrays with perfect autocorrelation and optimally low cross-correlation between all p family members. Now, we examine the cross-correlations between arrays from different families.

3.1 Extending the family of perfect $p \times p$ arrays

Consider family A of p $p \times p$ pn arrays, for $p = 4k–1$, derived from a Hadamard matrix. The array of optimally low cross-correlation values that is obtained by a periodic correlation between any two members from A, for example $B_{ij} = A_i \otimes A_j$, is itself a scaled pn array. This was noted in [6] and is easy to prove using Fourier transforms or finite field theory. Dividing the correlation array B_{ij} by p restores the ± 1 values.

Periodic cross-correlation of a pair of nD arrays is equivalent to multiplying their nD Fourier transforms (1). 1D projections of the FRT tile all non-zero frequencies in the nD Fourier space exactly once, because of the discrete central slice theorem and the prime array size. Hence, performing an nD correlation is equivalent to taking the dot product of the 1D FT of matching 1D projections in FRT space.

As pn arrays are always spectrally flat, this dot product only toggles the sign of the matching projection in the product array where the input array projection signs were different. As exactly half of each of the projection signs were originally matched and half were different, their product can flip exactly and only half of the signs and hence the correlation result is another perfect array. The new pn array B_{ij} is made by correlating the i^{th} and j^{th} arrays of A:

$$B_{ij} = \frac{1}{p} A_i \otimes A_j \tag{4}$$

or, in the Fourier domain,

$$F\left[B_{ij}\right] = F[A_i] \times F\left[A_j\right] \tag{5}$$

In (5), $\boldsymbol{F}$ denotes the 1D DFT of each of the $p + 1$ FRT projections of a 2D array.

The in-place dot-product over all spatial domain elements of any two zero-aligned 2D $p \times p$ perfect arrays is an alternative but equivalent and simple generating method (i.e. $C_{ij} = A_i \cdot A_j$). The product of in-place array elements also flips exactly half of the opposite signs within the two input arrays to produce a distinct pn array. We can show that, for all $i \neq j$, $Bij = C_{ij}$.

The full set of cross-correlations between all p members inside a perfect family of $p \times p$ arrays will generate a set of $p(p\text{-}1)/2$ perfect arrays. A new family, $\boldsymbol{B}$, of size $p\text{-}1$, of arrays with perfect autocorrelation and optimal cross-correlations can be selected from this set as $B_i = A_i \otimes \{A_1 \ldots A_j \ldots A_p\}$ for $j \neq i$.

The family B is disjoint from family A, meaning that there is no array A_i from A that is replicated as B_i in B. The cross-correlation between any pair of member arrays of family B is optimally low, because $A_i \otimes A_i = 1$, as A_i has perfect autocorrelation (a delta function) and $A_j \otimes A_k$ has optimal cross, because A_j and A_k are, by design, all maximally different arrays. Thus

$$\begin{aligned} B_j \otimes B_k &= \left(A_i \otimes A_j\right) \otimes \left(A_i \otimes A_k\right) \\ &= \left(A_i \otimes A_i\right) \otimes \left(A_j \otimes A_k\right) = A_j \otimes A_k \end{aligned} \tag{6}$$

Of special interest here is the spectrum of cross-correlation values that are possible between the two original arrays, A_i and A_j and their product B_k (i.e. $C_m = A_i \otimes B_k$ and $C_n = A_j \otimes B_k$). Of the $p(p\text{-}1)$ possible cross-correlations between members of $C = A \otimes B$, we find that, for all but very low values of p, the spectrum of individual cross-correlations, C_m and C_n, turns out to be either optimal or very close to optimal.

Such families of arrays are ideal for watermarking, where multiple arrays are embedded in the host media, for higher data payload, increased security or lower perceptibility. The cross-correlation between arrays embedded in the same media must be optimal to enable them to be extracted by correlation. The cross-correlation between arrays embedded in different host media can be relaxed slightly, because higher cross-correlation can only increase the probability of false positive detection. A small number of false positives are insufficient to cause mistaken identity of the host media. Also the assignment of arrays can have an error correction feature built in.

For families of $p \times p$ arrays, the spectrum of cross-correlation values $\boldsymbol{C}$ between family A and its derived family B was computed. The distribution within $\boldsymbol{C}$ turns out to be fixed for each p, independent of how family A was generated. The array that results from the lowest (optimum) possible cross-correlation has a center value of 0 and is elsewhere an equal distribution of $\pm p$.

The second lowest cross-correlation array has a center 0 and off-peak array values of $\pm$ $(p–4)$, because two pairs of projections from each array have their signs toggled. The third lowest correlation array has off-peak values of $\pm$ $(p\text{-}8)$, the fourth $\pm$ $(p\text{-}12)$. The i^{th} lowest correlation has a center 0 and off-peak array values of $\pm$ $(p–4(i\text{-}1))$. At correlation level $i = (p + 5)/4$, the off-peak array values all become -1. At this point, the cross-correlation reaches the same value as the autocorrelation and the two spatial domain arrays must be identical, up to a change of sign.

Table 1 shows the distribution of correlation results for $C = A \otimes B$, for *4k-1* primes $p = 7$ to *127*. The spectrum of cross-correlation values within C has optimally low values on the left, and the cross-correlation value worsens with each column to the right.

For $p = 7$, the entry in the third level of correlation values in Table 1 shows that 14.3% of the cross-correlations are worst case, generating identical copies by producing perfect autocorrelations. For $p = 7$, these worst-case cross-correlations occur at the third lowest level, $i = 3$, where all off-peak values change by $4(3–1) = 8$, so all cross-correlation values are either $+7 - 8 = -1$ or $-7 + 8 = -1$. Constructing low cross-correlation arrays for $p = 7$ by this mechanism appear to have limited use.

Note the very compact form of these distributions, and the difference pattern for the primes that are $8k \pm 1$or $8k \pm 3$. For some primes (e.g. $p = 23$ and *31*), the number of optimally low cross-correlations (level 1) forms a high percentage of all possible cross-correlations, and the next most frequent correlations are level *3* values. For $p = 23$, statistically 0 entries occur at and beyond the *4th* lowest level of cross-correlations. The worst case result for cross-correlations at $p = 23$ lies at the *7th* level $(7 = (23 + 5)/4)$. Cross-correlation between any array chosen from A and B here is either optimally low, or very close to optimally low.

As p increases, the sharp cut-off in the distribution of correlation values becomes more statistically certain: for

Table 1 Distribution of cross-correlation values between the p members of two $p \times p$ arrays

7	**0.52**	0.452	*0.03*	0	0	0							
11	0.432	**0.488**	0.08	*0.002*	0	0	0						
19	0.344	0.477	0.16	0.022	0	*0*	0	0					
23	0.316	0.464	0.18	0.036	0	0.0001	*6E-06*	0	0				
31	0.276	0.435	0.21	0.064	0.01	0.001	5E-05	9E-07	*0*				
43	0.236	0.396	0.24	0.098	0.03	0.005	0.0006	0.0001	0	0			
47	0.228	0.386	0.24	0.106	0.03	0.0075	0.0013	0.0001					
59	0.204	0.357	0.24	0.127	0.05	0.0154	0.0033	0.0005	0	0			
67	0.191	0.342	**0.24**	0.136	0.06	0.0211	0.0058	0.0012	2E-04	4E-05	5E-06		
71	0.186	0.334	0.24	0.14	0.06	0.0243	0.0071	0.0017	3E-04	0	0		
79	0.177	0.321	0.24	0.146	0.07	0.0301	0.01	0.0027	6E-04	0.00011	1E-05		
83	0.172	0.315	0.24	0.149	0.08	0.0337	0.0115	0.0034	0.008	0.00136	2E-04	2E-05	
103	0.155	0.289	0.23	0.157	0.09	0.0467	0.0201	0.0072	0.002	0.00061	1E-04	0.0002	1E-05
127	0.14	0.264	0.22	0.161	0.1	0.06	0.0302	0.0135	0.005	0.0018	6E-04	0.0001	3E-05
131	0.138	0.261	0.22	**0.162**	0.11	0.0626	0.0315	0.0144	0.006	0.002	6E-04	0.0002	6E-04
139	0.134	0.253	0.21	0.162	0.11	0.0645	0.0349	0.0166	0.007	0.0026	1E-04	0.0003	6E-04
151	0.128	0.244	0.21	0.159	0.11	0.0718	0.0395	0.0189	0.009	0.0035	0.001	0.0005	2E-04
199	0.112	0.217	0.19	0.157	0.12	0.0832	0.0537	0.0325	0.018	0.0091	0.004	0.0017	6E-04

Family B is made from the cross-correlation or dot products of A. The strength of the correlation is optimally low in the second left column and increases with each column to the right. The entries in italics mark the level at each p where the cross value becomes that of a perfect periodic autocorrelation. The sum of the entries on each row is normalized to 1. Bold entries mark the array size p where each correlation level has maximum frequency

$p = 31$, 0 events are observed at and beyond level *4*. The maximum statistically observable periodic cross-correlations have low values. For $p = 31$, cross-correlation are seen up to level *3*, where $MF_p = 0.066$, while the optimally low (level *1*) cross-correlation has $MF_p = 0.001$. The peak periodic autocorrelation MF_p is $31^2\text{-}1 = 960$, this occurs at level *9*.

Choosing a sensible value for p means we can then generate many perfect auto families B from a perfect family A and be confident that the cross-correlations between A and B will be asymptotically close to optimally low. Within family A or within family B, the array cross-correlations are exactly optimally low, and A and B are disjoint families.

3.2 Generalized array families, p = 4k-1

We next consider the shape of the distribution of cross-correlations between a perfect family A and a second, independently generated perfect family, A'. We can generate independent families by choosing a different Hadamard matrix to change the maximally different pattern of signs for the 1D FRT projections from which each family of arrays is built. The number and structure of possible unique and inequivalent Hadamard matrices is an ongoing research issue, and those numbers vary strongly with the matrix size 9.

However, an equivalent Hadamard matrix is easily formed by a random shuffling of any existing Hadamard matrix rows (or columns), provided that no row (or column) ends up back in its original place. The balance of signs is preserved, but their ordering across the rows is different. These shuffled versions are called equivalent Hadamard matrices.

There are $p!! = (p\text{-}1)(p\text{-}3)(..)(5)(3)(1)$ ways to do these shuffles for any $(p + 1)*(p + 1)$ Hadamard matrix. This number rises very rapidly with p: for $p = 7, 11, 19$ and 23, $p!!$ is of order *105*, *10,395*, 6×10^8, and 3×10^{11} respectively. Each of these shuffled variations makes a Hadamard matrix from which a $p \times$ p perfect family A can be built.

We computed the distribution of all possible cross-correlations between large numbers (several hundred) of pairs of families $C = A \otimes A'$, where each family of p $p \times p$ arrays was produced by a random shuffling of a particular Hadamard matrix. We repeated this process for $p = 4k\text{-}1$ primes between *7* and *199*. Table 2 shows that shuffling the generating Hadamard matrix produces disjoint sets of families that have near optimal cross-correlations between all family members.

The worst-case cross-correlations (statistically) reach towards the autocorrelation value (at level $i = (p + 5)/4$) at which results in exact copy or sign-matched arrays) only for $p < 23$.

The bold entries in Table 2 (in the columns for cross-correlation levels $i = 1$ to *4*) mark the primes p at which the maximum frequency occurs for each cross-correlation level.

Many different families of p $p \times p$ arrays can be built that have perfect autocorrelation and optimal cross-correlations within each family. More significantly, with a very high probability, the cross-correlation between any array chosen from one family and any array chosen from a second family will be asymptotically low, particularly for $p > 23$.

Shuffling a Hadamard to produce equivalent Hadamard matrices is a statistical process. We want to find equivalent Hadamard matrices that apply a similar number of toggles to the elements of the arrays. The number of toggles can be counted by summing the dot product of the 'parent' Hadamard matrix (HP), with its shuffled version, the 'child' (HC). Any single shuffle always produces $\pm M$ toggle differences with respect to its parent for an $M \times M$ equivalent Hadamard matrix.

This result explains the fixed spectrum of cross-correlation values observed in Table 1, between the p members of a $p \times p$ family A and the $(p\text{-}1)$ members of family B made from the dot product or cross-correlations of the arrays in A. Family B is the result of a child Hadamard HC produced by one shuffle of the parent Hadamard HP of family A.

The spectrum of values in Table 2 arises from cross-correlations between unconstrained equivalent Hadamard matrices that generate unconnected families. A parent Hadamard matrix, HP, is randomly shuffled to produce a child equivalent Hadamard matrix, HC1, giving rise to family A_1. Then the same parent HP is randomly shuffled again to produce another child equivalent Hadamard matrix, HC2, giving rise to family A_2. The cross-correlations between these families, $A_1 \otimes A_2$ drives the distributions of Table 2.

We examined the variation in the number of toggles for different equivalent Hadamard children HC from shuffling HP. We computed the absolute value of $\Sigma(\mathrm{HC1} \cdot \mathrm{HC2})$ for up to 10^8 random shuffles and tabulated these results, as shown in Table 3. For $p = 4k\text{-}1$, the number of toggle differences between child-to-child Hadamard matrices is quantised into discrete steps as $i(p + 1)$, where, statistically, $0 \le i \lesssim 9$.

Consequently, we can select, from a random set of equivalent Hadamard matrices shuffles that remain closest to the parent child set. This ensures that families are more likely to contain members whose cross-correlation values with other families will be close to optimally low. Statistically, the effect is only important for low values of p, where the spread between optimal and worst correlation levels is smallest. For larger primes, it is hard to not produce low correlation $p \times p$ sets by random shuffling of the parent Hadamard, as the gap between lowest and worst cross-correlation values is so large, and *91%*

Table 2 Distribution of cross-correlation values between family *A* and *p p* × *p* equivalent Hadamard variant

p = 4k-1	Frequency of increasing cross-correlation level									
	1	2	3	4	5	6	7	8	9	10
7	**0.5182**	0.4519	*0.0299*	0	0	0				
11	0.4324	**0.4879**	0.0774	*0.0023*	0	0	0			
19	0.3443	0.4769	0.1556	0.0222	0.001	*0*	0	0		
23	0.3158	0.4639	0.1812	0.0358	0.0032	0.0001	*6E-06*	0	0	
31	0.2755	0.4354	0.2135	0.0635	0.011	0.001	5E-05	9E-07	*0*	
43	0.2356	0.396	0.2381	0.0979	0.0275	0.005	0.0006	0.0001	0	0
47	0.2283	0.3857	0.2373	0.1064	0.0335	0.0075	0.0013	0.0001		
59	0.2041	0.3571	0.2416	0.1271	0.0508	0.0154	0.0033	0.0005	0	0
67	0.1913	0.3419	**0.2417**	0.1361	0.0606	0.0211	0.0058	0.0012	0.0002	4E-05
71	0.1862	0.3342	0.2416	0.1397	0.0648	0.0243	0.0071	0.0017	0.0003	0
79	0.1766	0.3206	0.2393	0.1464	0.0736	0.0301	0.01	0.0027	0.0006	0.0001
83	0.1721	0.3148	0.2386	0.1486	0.0764	0.0337	0.0115	0.0034	0.0079	0.0014
103	0.1546	0.2888	0.2302	0.1567	0.0924	0.0467	0.0201	0.0072	0.0024	0.0006
127	0.1404	0.2637	0.2189	0.1614	0.1042	0.06	0.0302	0.0135	0.0052	0.0018
131	0.1375	0.2606	0.2166	**0.1622**	0.1059	0.0626	0.0315	0.0144	0.0058	0.002
139	0.1342	0.2531	0.2147	0.1618	0.109	0.0645	0.0349	0.0166	0.0072	0.0026
151	0.1278	0.2438	0.2109	0.1586	0.114	0.0718	0.0395	0.0189	0.0094	0.0035
199	0.1118	0.2168	0.1927	0.1565	0.119	0.0832	0.0537	0.0325	0.0178	0.0091

The correlation is optimally low in the second left column and increases with each column to the right. The entries in italics mark the level at each p where the cross-correlation becomes an autocorrelation. The sum of the entries on each row is normalized to 1. Bold entries mark the array size p where the maximum frequency for level i occurs

of random shuffles have either *0* or ± *(p + 1)* toggle differences.

3.3 Generalized array families, p = 4k + 1

The distribution of cross-correlation values for pn arrays drops so sharply in Table 2, that it is worth considering what happens for pn arrays where the primes *p* are *4k + 1*. We can make individual *p* × *p* arrays for *p = 4k + 1* that have perfect autocorrelation, but far from lacking Hadamard matrices, not even one pair of *4k + 1* arrays can be made with optimum correlation. However, a large number $^{p+1}C_{(p+1)/2}$ of perfect autocorrelation arrays can be made by pseudorandom inversion of the signs of half of the 1D projections, and we can examine the distribution of cross-correlations for these sub-optimal cases. The results are surprisingly similar to the *p = 4k-1* case. The distribution of cross-correlations is more complex, as there are no orthogonal binary sequences of length *4k + 2*. As the cross-correlations are no longer equally balanced in sign, the center cross-correlation value is

Table 3 Frequency distribution for the number of sign changes between equivalent Hadamard matrices

p	0	1	2	3	4	5	6	7	8
19	0.367167	0.54284	0.067909	0.017541	0.003751	0.000664	0.00011	0.000017	0.000001
23	0.36764376	0.54392703	0.06669008	0.01737125	0.00362844	0.00063027	0.00009564	0.00001197	0.00000136
31	0.368055	0.545614	0.065292	0.016755	0.003587	0.000598	0.000084	0.000014	0.000001
71	0.367799	0.548824	0.063183	0.0162	0.003383	0.000518	0.000077	0.000015	0.000001
127	0.367731	0.550441	0.062346	0.015706	0.003159	0.000544	0.000061	0.000011	0.000001
199	0.36784095	0.55094903	0.06192199	0.01555615	0.00312504	0.0005226	0.00007395	0.00000912	0.00000103
307	0.367281	0.551637	0.061785	0.015528	0.003128	0.000551	0.000082	0.000007	0.000001
419	0.368032	0.551334	0.0616	0.015369	0.00306	0.000521	0.000072	0.000011	0.000001
499	0.367942	0.551026	0.061757	0.015476	0.003181	0.000536	0.000075	0.000007	0

The sum of the entries on each row is normalized to 1. For p = 23, at level 2 there are 3 × 24 = 72 sign changes between equivalent Hadamard matrices and this happens for 6.67% of shuffles. Note that the most frequent result, for all p, is when the number of toggles is the same as for a parent to child shuffle (level 1, which happens in about 55% of all shuffles) not 0, as it is for $p = 4k\text{-}1$. The off-peak entries for $p = 4k + 1$ come from arrays with unbalanced pairs of projections hence the cross-correlation values can be ± *(p-2i)* apart, leading to a wider range of cross-correlation values. The distribution of these values falls to 0 at a similar rate to *4k-1* primes. There is a penalty in building families of 2D perfect arrays from $p = 4k + 1$ primes, but it is not prohibitive, as demonstrated in Section 3.

Figure 3 shows one member A_i from a family of 5 5×5 arrays, the 5×6 set of 1D FRT projections used to make this array and the pseudo-Hadamard 6×6 matrix used to generate a family of perfect arrays. All $4k + 1$ pseudo-Hadamard matrices generated in this way have balanced off-diagonal entries of ±*2* (apart from the trivial leading row and column). The cross-correlations between family members for $p = 4k + 1$ arrays are all next-to-optimal (optimal values in Fig. 3 (*f*) would be ±*5* and *0* center).

3.4 Correlations between extended families

Cross-correlations between two extended families of arrays were examined to verify the construction method developed in Section 2. Two independent extended families (*A* and B) of 31×31 arrays were generated. Each extended family comprised *16* sets of *31* perfect autocorrelation, optimal cross-correlation, making a total of $16 \times 31 = 468$ 2D arrays per family. Each $31 \times 31 \times 468$ array could uniquely watermark every frame in *16* s of *30* fps video.

There are *246,016* cross-matched possible between members of family *A* with members of family *B*. Table 4 gives the result observed for two pairs of such families, the first pair generated by random selection of equivalent Hadamard matrices. The second pair had the equivalent Hadamard matrices selected to have matching number of sign toggle differences. This was repeated for pairs of extended families containing *121* 11×11 arrays. The results in Table 4 are consistent with the statistical distributions in Table 2.

4 Watermark embedding and extraction

Imperceptible spread-spectrum watermarks are typically embedded by adding a conveniently scaled watermark value to the raw data or in a transform domain. Some transforms are more useful than others, especially if they make use of the masking properties of the human visual system or the human auditory system. This makes embedded watermarks less perceptible. Also, embedding in some transform domains makes the watermark more compatible with common compression algorithms and consequently, more robust. A typical embedding scheme for images is illustrated in Fig. 4.

The watermark data is embedded as cyclic shift of the embedded array. For example, for $p = 257$, an array can contain two ASCII characters. We mark our images and video in the Luma channel only. This is because some images or video are greyscale only. Also, the watermark should survive conversions of colour format and Luma is the most robust.

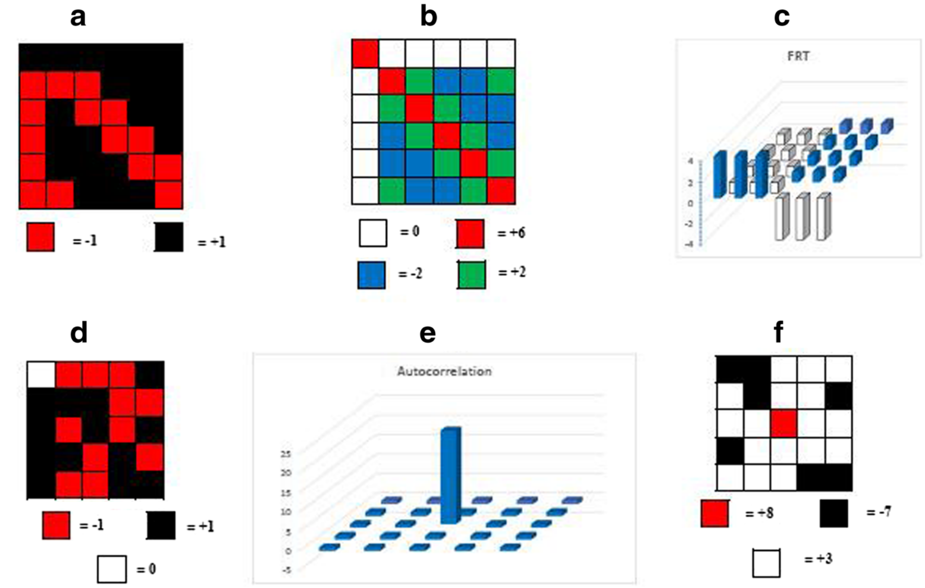


Fig. 3 **a** The 6×6 pseudo-Hadamard H_6. **b** The matrix product $H_6{*}H_6^T$ is quasi-orthogonal (c.f. Fig. 2a, b). **c** A set of six spectrally flat 1D FRT projections for $p = 5$. Here, the pattern of signs for each 1D projection is fixed by row 2 of (**a**). Four other distinct FRTs can be built using rows 3–6 of (**a**). **d** The inverse FRT of (**c**) is a perfect pn array. Each of the 5 5×5 arrays exhibits perfect autocorrelation, as shown in (**e**), but the 10 cross-correlations between these arrays, (**f**), are next-to-optimal (c.f. Fig. 2f)

Table 4 Cross-correlation statistics, ordered by increasing levels of cross-correlation value, between two families that each contain

Level	1	2	3	4	5	6	7	8	9
(a)	67,614	107,373	52,510	15,512	2715	279	13	0	0
(b)	67,444	107,512	52,528	15,548	2695	276	9	0	0
(c)	6284	7193	1138	26					
(d)	6269	7223	1117	32					

a 468 random 31 × 31 arrays, *b* 468 matched equivalent Hadamard 31 × 31 arrays, *c* 121 random 11 × 11 arrays, *d* 121 matched equivalent Hadamard 11 × 11 arrays

Watermarks are typically extracted by correlating a template of the mark with the watermarked data. The extraction process is illustrated schematically in Fig. 5.

The process of correlation is described in equation 1 Watermarks also depend on the dimensionality of the data, or the transform domain. Video can be considered as 3D as in [13] or 2D if individual frames are marked independently. Watermarks may be used for different purposes—copyright protection or proof of ownership or authentication, audit trail, proof of tampering, etc. Consequently, watermarks can be classified as robust, fragile or semi-fragile [23]. An ideal robust watermark is one that resists all distortions and other attempts to remove it. A fragile watermark, on the other hand, can act as a checksum, so that any change will render it unreadable. This would apply to authentication. Many semi-fragile image watermarks are still recoverable after change, but will reveal the regions of an image that were changed. This might be useful, for example, in newsreel photographs—indeed, digital cameras now exist which embed such a watermark automatically. Many of the watermarking algorithms initially published required that the original data be available to the detector. It would typically be subtracted from the watermarked data to reveal the watermark itself. These are referred to as private watermarks. Watermark detectors that do not need the original have been called blind detectors.

The extraction process can be augmented by matched filtering for the watermark. This is because, ideally, the correlation of the watermark should be a delta function. A Laplacian kernel is a good example. 2D and 3D kernels are shown below

$$L_{2D} = \frac{1}{8}\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

$$L_{3D} = \frac{1}{26}\begin{bmatrix} -1 & -1 & -1 & & -1 & -1 & -1 & & -1 & -1 & -1 \\ -1 & -1 & -1 & \vdots & -1 & 26 & -1 & \vdots & -1 & -1 & -1 \\ -1 & -1 & -1 & & -1 & -1 & -1 & & -1 & -1 & -1 \end{bmatrix}$$

It is also possible to suppress the image contribution to the correlation by performing averaging functions.

The objective perceptibility of a watermark is usually measured by PSNR (peak signal to noise ratio), which is defined for images

$$\mathrm{PSNR} = 10log_{10}\frac{\mathrm{MAX}^2\mathrm{I}}{\mathrm{MSE}}$$

Where MAXI is the maximum possible value of the image and MSE is the mean square error. The PSNR is dependent on the video data, and the compression and decompression implementation. This is because intra-frame and inter-frame compression introduce different

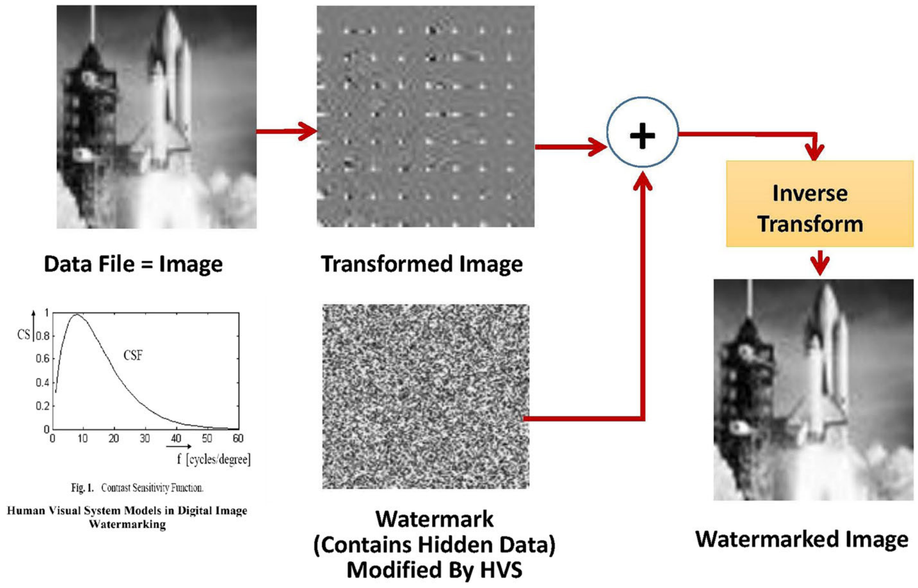

Data File = Image
Transformed Image
+
Inverse Transform
CS
1
0.8
0.6
0.4
0.2
0
0
10
20
30
40
50
60
CSF
f [cycles/degree]
Fig. 1. Contrast Sensitivity Function.
Human Visual System Models in Digital Image Watermarking
Watermark (Contains Hidden Data) Modified By HVS
Watermarked Image

Fig. 4 Watermark embedding scheme

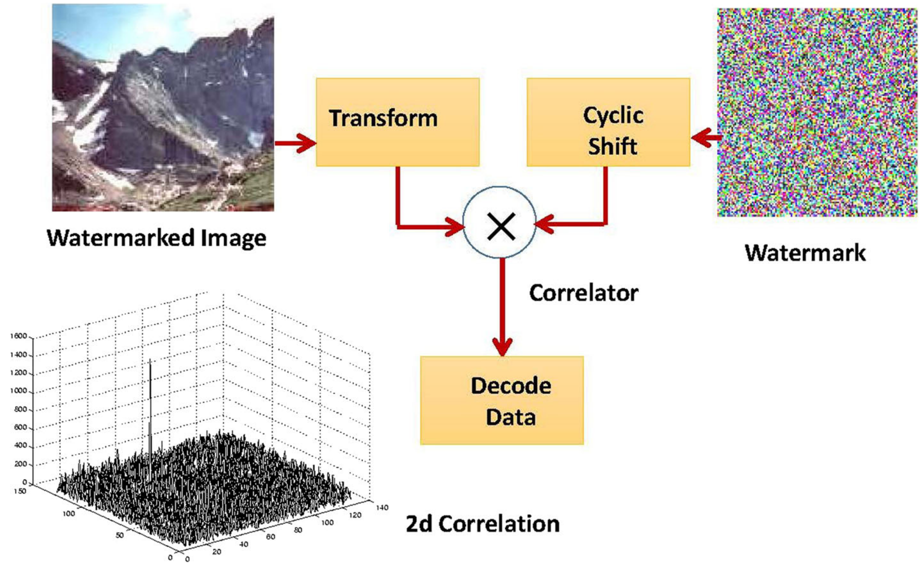


Fig. 5 Watermark extraction process

artefacts. In our experiments, we used H264 compression. We found that an MS Windows-based decompression introduced variable frame delays before stable acquisition of motion vectors, and this caused fluctuations in PSNR measurements. Analysis of the PSNR of our video watermark will be undertaken elsewhere. Our scheme affects a variable proportion of all pixels, by adding or subtracting an LSB. Consequently, the PSNR for uncompressed video is always above 48.13 dB. Not all the image frames are marked, to reduce motion vector artefacts. Each image frame is marked in blocks commensurate with H264 compression, with some blocks remaining unmarked. Also, the block boundaries are graded, to reduce intra-frame compression artefacts. A pseudorandom selection of about 50% of pixels within a block are marked, to increase effective PSNR to above 51 dB. However, as Fig. 6. shows, the effects introduced by the watermark in compressed video are commensurate with those introduced by compression artefacts alone. It was not known if the choice of marking blocks commensurate with compression blocks could cause the portions of our arrays to appear in compressed video. This would be of concern, because it could reveal our array constructions to an attacker. Fortunately, as Fig. 6 demonstrates, that is not the case.

This is as expected, because our watermark is confined to LSB modification. We plan to implement a similar strategy to our video watermarks as to our still image version (https://watermarking-print-and-scan.firebaseapp.com/opening). This involves the embedding of two watermarks—a robust one in pixel domain to act as proof of watermarking and registration/synchronization for a frequency domain mark, which is fragile. Detection of the robust one alone indicates tampering, detection of both is proof of authenticity. The original concept of the dual mark is explained for audio watermarks in [24].

There are few genuine video watermarks. The ATSC is effectively relying on the audio component of video to provide copyright protection, on the assumption that video is always accompanied by audio. Of course, audio can be substituted, dubbed or otherwise remixed. The video watermark in the standard is a token—confined to the first two lines of the video, which are made inaccessible to the user. The principal objective of such a watermark is to provide content identification and monitoring for set top boxes.

> *The video watermarking technology specified herein involves modulation of the luma component of video within the top two lines of active video in each video frame. Two encoding options are offered, one providing a watermark payload of 30 bytes per video frame (a "1X"version), and the second "2X" version offering double that capacity. Visibility of this video watermark is not anticipated to be an issue because ATSC 3.0-aware receivers are expected to be designed with the knowledge that the top two lines of active video may include this watermark, and will thus avoid displaying (by any means desired). The majority of HD TV display systems in use at the time of publication operate by default in an "overscan" mode in which only the central ~95% of video lines are displayed. Thus, if watermarked video is delivered to a non-ATSC 3.0-aware receiver, the watermark would not normally be seen* [25].

Fig. 6 Left unmarked frame, right marked frame

There are companies offering video watermarks which are not separable from the video stream. Most if not all rely on a concept introduced by Digimarc®, a pioneer in the commercial watermarking area, embodied in patents such as [26, 27]. Typically, these embed a pseudorandom sequence generated by a random number generator (RNG) in the video stream or a transform of it. This stochastic method has a fundamental limitation. There is no guarantee that any pseudorandom sequence generated by a RNG is unique. Licensees of the Digimarc method test pseudorandom sequences using cross-correlation to ensure that their watermarks are sufficiently distinct.

By contrast, our algebraic method guarantees that all our watermarks are unique by construction. Families of sequences or arrays have correlations whose maximum value has a lower bound, which depends on sequence length/array size (L), family size (F) and type of symbol used (alphabet). For binary and other roots of unity alphabet, the Sidelnikov bound applies [18], while for general complex valued alphabet, the Welch bound is appropriate [17]. These bounds are complicated, but a graphical interpretation is possible as shown in Fig. 7.

Our arrays meet the bound, as indicated in the figure. Consequently, our watermarks attain the lowest possible false detection rates because of the low cross-correlation, and the lowest missed detection rate, because of the off-peak autocorrelation being −1. The low cross-correlation also permits the embedding of multiple arrays in the same media, to increase data payload.

Of course, false or missed detection can still occur, due to high cross-correlation with image data, but it is very unlikely that the image resembles our arrays, and anyway the image content is outside our control. Our online watermarking app (https://watermarking-print-and-scan.firebaseapp.com/opening) has an image analysis package which tests the suitability of an image for watermarking, and images with high cross-correlation with our arrays are rejected. As mentioned, our image and audio watermarks are composed of two components—spatial and Fourier domain for images and time domain and frequency domain for audio. It is extremely unlikely for the media to resemble our arrays in both domains simultaneously.

Also, there are some advantages to the stochastic watermark using a RNG. It is more difficult to reverse engineer, because the search space for the RNG seed can be prohibitive. We have developed a method of combining our arrays constructed using Finite Field algebra with stochastically generated ones, to match the security of the RNG method. This will be described in a future paper.

Our algebraically generated families of arrays are scalable in size and available for all even dimensions. Video can be considered as three dimensional, with time, or frame number taking the role of the third dimension. The audio stream can be folded and appended as a fourth dimension. Such a 4D multimedia watermark can be useful in determining if the video, audio or audio sync has been tampered with. The authenticity of audio-visual evidence, including audio synchronization has been a critical and contentious issue in criminal trials and investigations, such as in (https://www.tamilsagainstgenocide.org/Docs/DublinTribunal/TAG-PPT-Extra-judicial%20Executions.pdf).

5 3D video watermarking

Online video piracy costs premium entertainment companies over $ 6 billion each year. SMPTE (Society of Motion Picture & Television Engineers) and ATSC (Advanced Television Systems Committee) have identified video watermarking and fingerprinting as a frontline defense against such piracy. Content monitoring is an additional benefit of watermarks and fingerprints, and

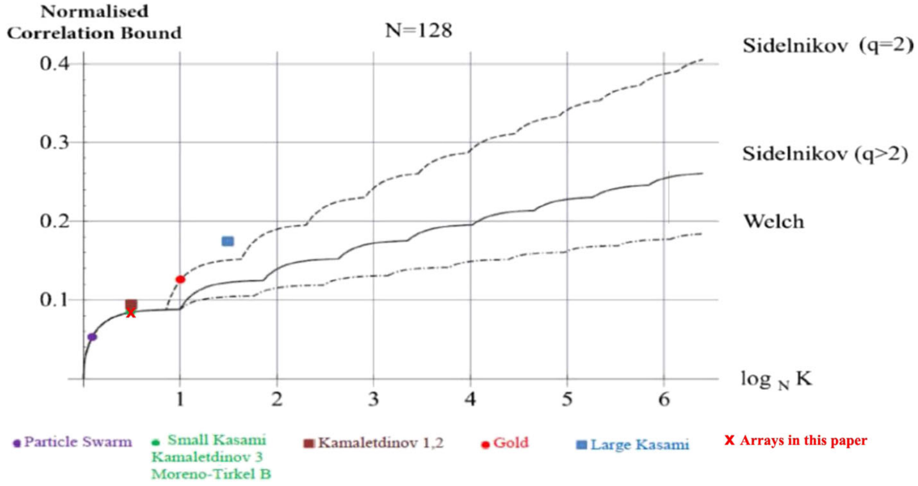


Fig. 7 Normalized correlation bound versus family size and alphabet (adapted from [30])

advertising companies are becoming aware of this as a means of offering directed advertising. SMPTE is in the process of developing a video watermarking standard through their competition to establish a standard for the movie industry '24 TB Open binding technology for persistent content identification in A/V essence'. ATSC is further down the track into adopting standard A/335 for video watermarks and A/334 for audio watermarks.

The anti-piracy and directed advertising objectives impose challenging requirements on the video watermark. The watermark must be imperceptible, and yet capable of extraction from a *2*-s video clip. Originally, the SMPTE watermark was required to be extracted from a *30*-s video clip. In response to the original *30*-s requirement, our group developed an embedding, extraction and analysis package for 3D video watermarking of H264 videos. Details of these will be presented elsewhere.

The arrays used for embedding and extraction were the two types described by the patent [28]. They were constructed using Finite Field algebra and were of the form $p \times p \times (p^2 - 1)$. $p = 17$ was chosen to be commensurate with the largest intra-frame macroblock sizes used in H264 compression. The long dimension was the frame index. For a *25*-fps rate, the complete array fills a time slot of *11.52* s, assuming no averaging is involved i.e. every frame is marked with a different plane of the 3D array. This is far too long for the current requirement of a *2*-s clip.

The large families of small cubic arrays presented in this paper solve this problem. Here, we show how we propose to do this. First, we present some results demonstrating the feasibility and effectiveness of our method of watermarking and use of arrays produced by our constructions. The first requirement of watermarking is imperceptibility. The left of Fig. 3 shows a frame from an unmarked video, while the right shows the same frame after marking with the array described above. The compression artefacts due to H264 processing in the left frame are not too dissimilar to the patterns due to the watermark, which are superimposed on the compression artefacts in the watermarked frame on the right. The frames were quite dark, and the intensity curves had to be adjusted to bring out the patterns. This resulted in a mismatch in contrast and hue. There were also H264 synchronization issues, due to variable buffer delays in Windows, which resulted in loss of sync between I frames within a GOP. All the same, Fig. 6. demonstrates that the watermark was unobtrusive or even imperceptible. The second requirement of watermarking is that the watermark can be successfully extracted from the media in which it was embedded with sufficiently low probability of missed or false detection. Figure 1 shows an unmistakable autocorrelation peak, which is clearly distinguishable from the cross-correlation with the video. These results were obtained without matched filtering.

Here, the array is of the form $17 \times 17 \times 288$ i.e. a volume of *83,232*. Analysis suggests that this is characteristic of a Peak/σ ratio (SNR) of *7.2*. Consequently, an off-peak autocorrelation, or a cross-correlation (RMS) of the order of *1/96.3* of the peak autocorrelation i.e. (*1936*) has insignificant effect on the probabilities of false or missed detection. This changes the SNR from *7.2* to *7.18*. This changes the probability of missed/false detection from *2.2E-12* to *2.54E-12*.

5.1 Implementation

We developed three modules to implement and test our watermarking scheme.

1 A watermark array generator.

This was developed using Wolfram Mathematica 10 ® and subsequently implemented in C++ and VHDL for ready transfer into FPGA's. The generator also produces other proprietary arrays and sequences for audio, image, video watermarking and for wireless communications, radar, UWB and MIMO systems. This generator is a general utility for evaluation of all these sequences and arrays. In terms of the specific implementation of the arrays described in this paper, it is trivial. It relies on a look-up table (LUT) of binary Hadamard matrices commensurate with the size of the image and applying the rows of such a Hadamard matrix to the projections of the array.

2 A video watermark embedding module using an Avisynth platform as illustrated in Fig. 8.
3 This user-friendly interface permits the user to embed chosen arrays in chosen video frames.

A screen capture of the GUI is shown in Fig. 9. The watermark embedding strength, persistence and the anticipated video compression are all user adjustable. Most of the controls in this screen are self-evident and are commonly used. This includes the REC, STOP, PAUSE, PLAY buttons, except that here the PLAY button also shows the extraction result of an embedded watermark as superimposed text, and the REC button also embeds a watermark.

4 A video watermark analyser module.

This displays frame by frame correlation of the watermarked video with the reference array and the statistics regarding the correlation peak height and its detailed statistics. This is also based on Avisynth platform. Examples of the analytics available from this utility are shown in Fig. 10.

5.2 Proposed cubical construction

We propose to use cubical arrays using $p = 37$. At *25* fps, this corresponds to *1.48* s assuming no averaging. This is in line with the SMPTE and ATSC standard being negotiated currently. The volume is *50,653*. It is expected that the cross-correlation behavior with the marked video will be similar to that described in Section 3.3.. Consequently, the Peak/σ ratio (SNR) is expected to be *5.6*. The corresponding probability of error is 6.6×10^{-7}. Let the effect of including an off-peak autocorrelation or cross-correlation with another array to affect the probability of error by less than *5%*. The RMS autocorrelation cross-correlation should be constrained to less than $\frac{1}{82.8}$ of the peak. This is commensurate with the array for $p = 31$ in Fig. 7, which shows a normalized RMS autocorrelation and cross-correlation of approximately *1%*.

5.3 Constructed versus random arrays

A random array of volume *83,232* has an autocorrelation with standard deviation $\sigma = 288.5$. A video can accommodate 3σ before any measured deterioration of the detection statistics is observed. For a random array, the probability of an autocorrelation exceeding 3σ is 0.00443. The cross-correlation is expected to behave in a similar manner. Our construction can use a spectrum which includes *3* levels before it runs out of family members. Each of our levels is of order σ. The number of our arrays with autocorrelation and cross-correlation bounded by 3σ is of order of *24* million. The probability that a RNG will deliver that number without exceeding

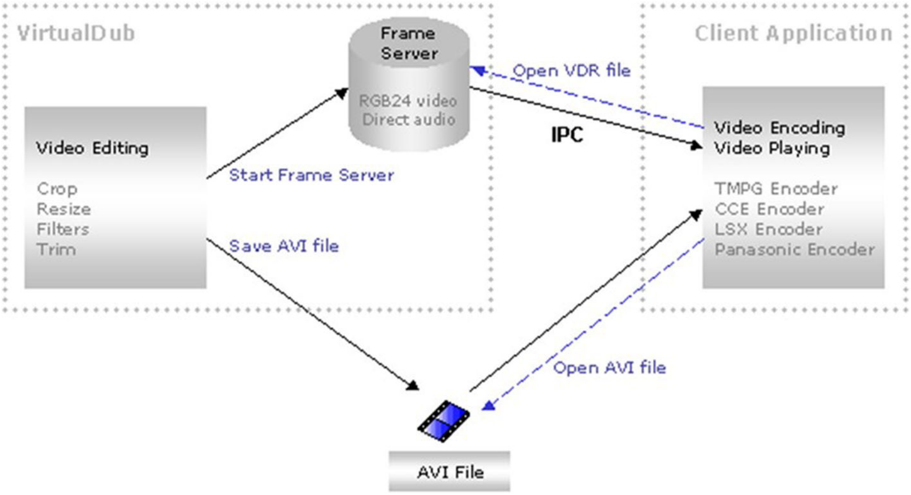
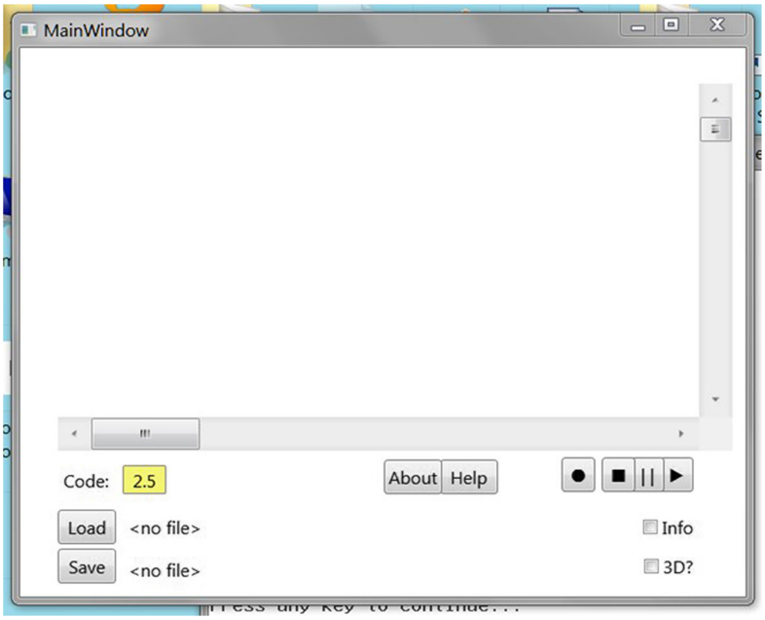


Fig. 8 The frame-serving process as implemented by Avisynth



Code: 2.5

Load <no file>

Save <no file>

□ Info

□ 3D?

Press any key to continue...

Fig. 9 Watermark Embedder and Player GUI

the threshold is of order $10^{-46,061}$ i.e. it is certain that it will fail. In fact, it takes *156* runs of a RNG for a *50%* probability of exceeding the threshold. The RNG users cannot 'filter' based on bad correlations, because they cannot perform that many correlations. By contrast, we can deliver nearly *24* million arrays by construction, with absolute certainty that they will not significantly degrade the probability of correct detection.

6 Conclusions

6.1 Conclusion

Previous work [6] presented a method to construct a family of $p \times p$ arrays that contained p members. These arrays have perfect autocorrelation and optimally low cross-correlation between any pair of family members. The array size p was a *4k-1* prime. This paper presents new methods to extend the size of these families of 2D arrays. Large families of arrays having this combination of properties are highly unusual. We first show how to produce disjoint families from an existing family, where each new family contains *p-1* members. We then show that multiple disjoint families, each of size p, can be constructed using equivalent Hadamard matrices. The members of these families all retain perfect autocorrelation and optimally low intra-family cross-correlation.

The inter-family cross-correlations are then shown to be next-to-optimally low. This means the distribution of periodic cross-correlation values shifts from full occupation of the lowest possible values to partial occupation of that level and partial occupation of the adjacent next-to-lowest levels. For aperiodic cross-correlations, the shift in this distribution has marginal statistical significance. We exploit the stability of these statistics to construct families of size np, where n can be of order p. The risk of creating accidental strong cross-correlations between the extended families of these arrays is exceptionally small for $p > 11$. We show that nearly optimal families of size p, with $p = 4k + 1$ prime, can be generated using pseudo-Hadamard matrices. These constructions make use of discrete projective geometry, Hadamard matrices and Finite Fields. We have prospective applications for these families of 2D arrays to watermark diverse forms of digital media.

6.2 Future work

The first idea used in this work is the reconstruction of a $p \times p$ array from sets of $p + 1$ discrete 1D FRT projections. Here, each 1D projection is spectrally flat i.e. perfect autocorrelation sequence. We could equally well substitute spectrally flat pseudo-noise sequences in place of the spectrally flat delta functions used here. The structure of these families of arrays, which we call phase-shifted arrays, has some significant differences from the extended families of arrays presented here. That work will be presented in a forthcoming paper.

We can also reconstruct 3D perfect arrays from sets of 1D perfect projections. The 3D approach requires $p^2 + p + 1$ spectrally flat 1D sequences. We can apply the 3D inverse FRT to create individual $p \times p \times p$ arrays that have perfect autocorrelation. However, to build a family of such arrays requires application of the second key idea in this paper: that the 1D sequences be maximally

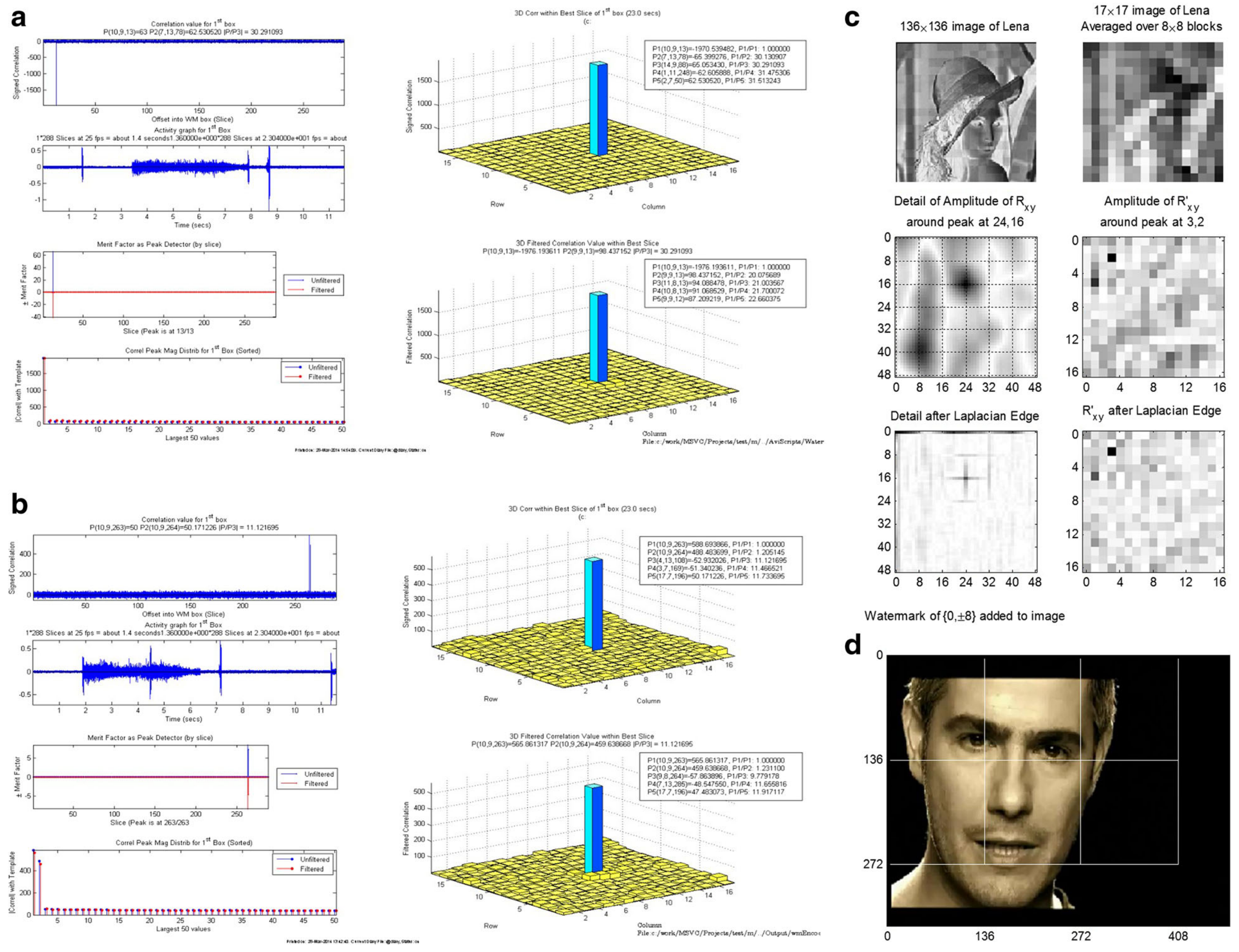

a
Correlation value for 1st box
Signed Correlation
Offset into WM box (Slice)
Activity graph for 1st Box
Time (secs)
Merit Factor as Peak Detector (by slice)
± Merit Factor
Unfiltered
Filtered
Correl Peak Mag Distrib for 1st Box (Sorted)
|Correl| with Template
Largest 50 values
3D Corr within Best Slice of 1st box (23.0 secs)
3D Filtered Correlation Value within Best Slice
Filtered Correlation
Row
Column
b
c
136×136 image of Lena
17×17 image of Lena Averaged over 8×8 blocks
Detail of Amplitude of R_xy around peak at 24,16
Amplitude of R'_xy around peak at 3,2
Detail after Laplacian Edge
R'_xy after Laplacian Edge
Watermark of {0,±8} added to image
d
0
136
272
408

Fig. 10 a Output screen from wmAvs before compression. *Left*—(i) correlation as a 1D stream. (ii) Activity graph of energy difference between adjacent frames and (iii)—merit factor for peaks in a 2D slice correlation. (iv) Sorted correlation peaks and noise. *Right*—(i) peak without Laplace filter. (ii) After Laplace filter. **b** Output screen after H264 compression. The sign of the correlation peak in (i) is due to a sync error. Slight noise increase in (iii) is due to compression loss. In (iv), the peak has split due to sync loss causing the correlation to straddle slices. **c** *Left*—(i) 136 × 136 marked image with each pixel replicated eight times in x,y to reduce compression damage. (ii) Cross-correlation with the template as an image. (iii) Laplacian of (ii). *Right*—Latent image is first reduced by 8 × 8 with, each block replaced by its average value. **d** A watermarked frame. The forehead is saturated, and the background is 0. The watermark is not added to those pixels

different in the Hadamard sense. As the number of 1D projections needed for 3D arrays is odd (for all primes), a family of $p \times p \times p$ arrays with optimally low periodic cross-correlations is not possible. We are continuing to examine the distribution of cross-correlation values obtained after using different methods of making a family of FRT arrays that are as different as possible. In 4D, where $p^3 + p^2 + p + 1$ discrete FRT projections are needed, the maximally different approach using Hadamard matrices again becomes possible for any prime size p.

The arrays constructed here have physical counterparts as finite 2D planes that contain various arrangements of atoms, where each atom has a quantized spin. The Bernasconi model [29] showed that 1D binary chains with maximal autocorrelation correspond to atoms arranged in their lowest energy levels (ground states). The real signed integer elements of the 2D arrays made here can be interpreted to be spins with different z-projected values that scale the strength of their interactions. The optimally low cross-correlation between pairs of arrays within a family can be considered to represent stacks of planes of atoms into 3D structures that have minimal energy.

Acknowledgements

The authors thank the School of Physics and Astronomy, Monash University for offering to cover publication charges. We also thank Dr. Ron van Schyndel and Dr. Nick Le Marshall for helping out with some of the figures.

Funding

This research received no specific grant from any funding agency in the public, commercial or not-for profit sectors.

Authors' contributions

The work was performed equally by both authors working as a team. Both authors read and approved the final manuscript.

Competing interests

The authors declare that they have no competing interests.

Author details

[1]School of Physics and Astronomy, Monash University, Clayton 3800, Australia. [2]Scientific Technology, 8 Cecil St, East Brighton, Melbourne 3187, Australia.

Received: 9 August 2016 Accepted: 10 February 2017
Published online: 20 February 2017

References

1. HD Lüke, Sequences and arrays with perfect periodic correlation. IEEE Trans. Aerosp. Electron. Syst. **24**(3), 287–294, (1988)
2. AZ Tirkel, CF Osborne, TE Hall, Image and watermark registration. Signal Processing J **66**, 373–383 (1997)
3. K Drakakis, R Gow, S Rickard, J Sheekey, K Taylor, On the maximal cross-correlation of algebraically constructed Costas arrays. IEEE Trans. Inf. Theory **57**(7), 4612–4621 (2011)
4. AZ Tirkel, CF Osborne, N Mee, GA Rankin, A McAndrew, Maximal connected sets—application to CDMA. Int. J. Digit. Analog. Commun. Syst. **7**, 29–32 (1994)
5. AZ Tirkel, *Cross-correlation of M-sequences—some unusual coincidences*. Fourth International Symposium on Spread Spectrum Techniques & Applications (IEEE ISSSTA'96, Mainz, 1996), pp. 969–974
6. A Tirkel, B Cavy, I Svalbe, *Families of multi-dimensional arrays with optimal correlations between all members*. Electronics Letters online. http://digitallibrary.theiet.org/content/journals/10.1049/el.2015.1046
7. O Phillipé, *Image representation for joint source-channel coding for QoS networks* (PhD Thesis, University of Nantes, France, 1998)
8. JP Guédon, Chapter 3, Section 3.2.2, p. 53, in *The Mojette Transform: theory and applications* (ISTE-Wiley, 2009). ISBN-10: 1848210809
9. KJ Horadam, *Hadamard Matrices and their applications* (Princeton University Press, 2007). ISBN-13: 978-0-691-11921-2
10. J Cox, J Kilian, FT Leighton, T Shamoon, Secure spread spectrum watermarking for multimedia. IEEE Trans. Image Process. **6**(12), 1673–1687 (1997)
11. N.J.A Sloane. http://oeis.org/A007299
12. Request for proposals open binding Of IDs to media. https://kws.smpte.org/kws/public/projects/project/details?project_id=284
13. S Blake, O Moreno, A Tirkel, Families of 3D arrays for video watermarking SETA 2014, Melbourne November 25 K.-U, in *Families of 3D arrays for video watermarking SETA 2014, Melbourne November 25 K.-U*, ed. by Schmidt, A Winterhof (SETA 2014, LNCS 8865, 2014), pp. 134–145
14. AZ Tirkel, GA Rankin, RM Van Schyndel, WJ Ho, NRA Mee, CF Osborne, *Electronic watermark*. DICTA 93, Macquarie University, 1993, pp. 666–673
15. R van Schyndel, AZ Tirkel, ID Svalbe, *A multiplicative color watermark* (IEEE-EURASIP Workshop on Non-Linear Signal and Imaging Processing, Antalya, 1999), pp. 336–340
16. SW Golomb, *Shift register sequences* (Aegean Park Press, 1967). ISBN-10: 0894120484
17. LR Welch, Lower bounds on the maximum cross correlation of signals. IEEE Trans. Inf. Theory **20**(3), 397–9 (1974)
18. VM Sidelnikov, On mutual correlation of sequences. Soviet Math Doklady **IT-12**(1), 197–201 (1991)
19. MJE Golay, The merit factor of legendre sequences. IEEE Trans. Inf. Theory **IT - 29**(6), 934–936 (1983)
20. L-J Weng, Decomposition of m-sequences and its applications. IEEE Trans. Inf. Theory **17**(4), 457–463 (1971)
21. F Matúš, J Flusser, Image representation via a finite Radon transform. IEEE T-PAMI **15**(10), 996–1006 (1993)
22. I Svalbe, *Near-perfect correlation functions based on zero-sum projections* (Proc. DICTA, Noosa). doi:10.1109/DICTA.2011.111L
23. RG van Schyndel, AZ Tirkel, ID Svalbe, TE Hall, CF Osborne, *Spread-spectrum digital watermarking concepts and higher dimensional array constructions*. First International Online Symposium on Electronics Engineering, 2000. http://www.techonline.com/osee (Presented online by R.. G. van Schyndel)
24. A Tirkel, T Hall, C Osborne, N Meinhold, O Moreno, *Collusion resistant fingerprinting of digital audio* (SIN 2011, Sydney, 2011), pp. 5–12
25. ATSC Standard, *Video watermark emission (A/335) Doc. A/335:2016*, 2016. http://atsc.org/atsc-30-standard/a3352016-video-watermark-emission/
26. KL Levy, *US 7,197,164 B2 Time-varying video watermark*, 2007
27. KL Levy, SK Decker, *US 7,020,303 B2 Feature-based watermarks and watermark detection strategies*, 2006
28. OM de Ayala, AZ Tirkel, Digital watermarking. US Patent **8**, 934–663 (2015)
29. AN Leukhin, EN Potekhin, A Bernasconi model for constructing ground-state spins system and optimal binary sequences. J. Phys. Conf. Ser. 613 (2015). doi:10.1088/1742-6596/613/012006
30. M Stular, S Tomazic, *Maximum periodic correlation of pseudo-random sequences in CDMA*. Electrotechnical Conference 2000, MELECON 2000, 10th Mediterranean, vol. 1, 2000, pp. 420–423