ScienceDirect

Hadamard Transform

A Hadamard transform is then used as a precoding to build packets of the same importance and the same energy.

From: Academic Press Library in Mobile and Wireless Communications, 2014

Related terms:

Equalizers, Qubit, Ancillary Qubits, Quantum Circuit, Superposition State

View all Topics

Learn more about Hadamard Transform

Raman and Infrared Microspectroscopy

Pina Colarusso, ... E. Neil Lewis, in Encyclopedia of Spectroscopy and Spectrometry, 1999

Hadamard transform imaging

Spatial encoding methods such as Hadamard transform imaging also can be used for the spectroscopic visualization of samples. More recently, developments in digital microarray technology will likely provide a convenient new approach for spatial encoding from the mid-infrared to the ultraviolet. In one common arrangement, the entire sample area is irradiated with wide-field, epi-illumination (Figure 4D). Part of the Raman signal emanating from the sample is blocked with a mask containing a series of apertures. The spatially filtered signal is focused on to an entrance slit of a monochromator, which disperses the signal across a two-dimensional array detector. The slit preserves one image axis while the Hadamard mask is used to encode the other axis. Subsequent measurements are carried out with the mask in different positions. Each measurement corresponds to the Raman signal from the unmasked points on the sample along one spatial axis over the entire spectral range of interest. The experiment is designed such that the number of independent measurements equals the number of points on the sample. The spatially dependent images are then converted to spectroscopic images through a Hadamard transform. Unlike the other methods mentioned in this article, Hadamard spectroscopic imaging systems are limited to research activities and are not yet commercially available.

equals the number of points on the sample. The spatially dependent images are then converted to spectroscopic images through a Hadamard transform. Unlike the other methods mentioned in this article, Hadamard spectroscopic imaging systems are limited to research activities and are not yet commercially available.

> Read full chapter

Transforms for Image and Video Coding

David R. Bull, in Communicating Pictures, 2014

5.3.2 The Walsh–Hadamard transform

The Walsh and Walsh–Hadamard transforms are simple but effective ways of compressing data. They have the significant advantage that the basic transform requires no multiplications, only sums and differences. While their coding gain is lower than transforms such as the DCT which we will consider shortly, they do find application in modern video compression systems, such as H.264/AVC, where they are used in intra-frame coding.

The discrete Walsh–Hadamard transform (DWHT) is obtained from a simple rearrangement of the discrete Hadamard matrix. The Hadamard matrix is an matrix with the property . Higher order matrices can be found by iteratively applying the following operation:

(5.19)

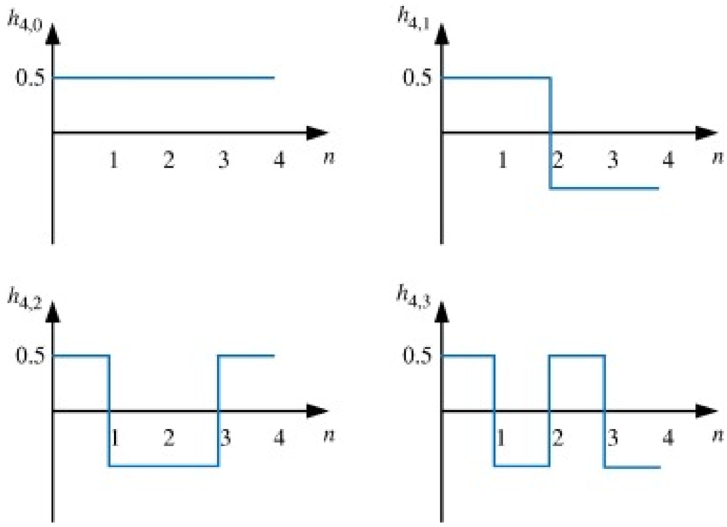
For example:

(5.20)

The DWHT is simply obtained from the corresponding Hadamard matrix by normalization and rearranging the rows in sequency order (i.e. in terms of the number of sign changes). Therefore the four-point DWHT is given by:

(5.21)

The basis functions for the 1-D DWHT are depicted in Figure 5.4.



$h_{4,0}$
0.5
1 2 3 4
n
$h_{4,1}$
0.5
1 2 3 4
n
$h_{4,2}$
0.5
1 2 3 4
n
$h_{4,3}$
0.5
1 2 3 4
n

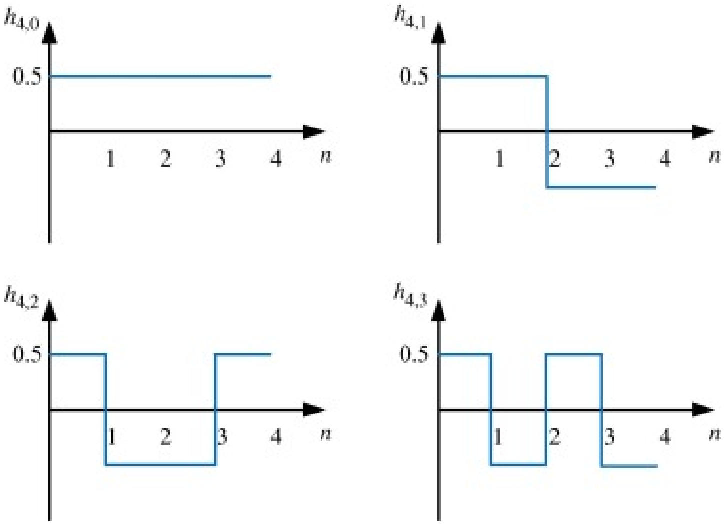


Figure 5.4. DWHT basis functions for .

In the case of the 2-D transform, the basis functions are again formed from the outer product of the individual 1-D basis functions. The basis functions for the 2-D DWHT are shown in Figure 5.5. **Example 5.3** 1-D DWHT Compute the 1-D DWHT for the data vector . **Solution** **Example 5.4** 2-D DWHT Compute the 2-D DWHT of the input image block S below: **Solution** Since the 2-D DWHT is separable, we can use equation (5.14). Letting where Then The energy compaction and decorrelation properties of the DWHT can clearly be seen in this simple example. Recall that, in the case of the original image block, the energy was distributed fairly uniformly across the block. After transformation, the data has been decorrelated horizontally and vertically and one dominant coefficient (top left) now contains some 93% of the energy. **Example 5.5** 2-D DWHT compression Taking the result of Example 5.4, set all small-valued coefficients (when) to zero and perform the inverse transformation. What is the PSNR of the reconstructed signal if the original signal was represented with a 6 bit word length? **Solution** Now with a 6 bit word length, the MSE = 3 and the PSNR of the reconstruction is:

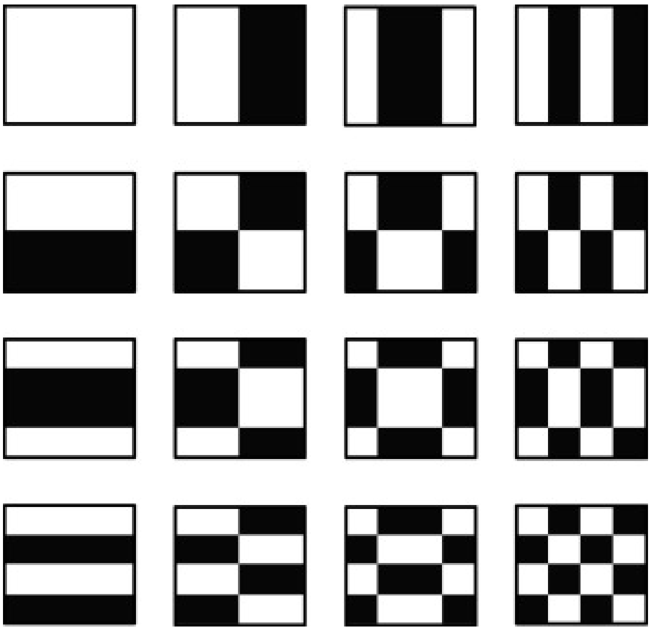


Figure 5.5. Basis functions for the 2-D DWHT.

Figure 5.5. Basis functions for the 2-D DWHT.

> Read full chapter

Applications in Data-Intensive Computing

Anuj R. Shah, ... Nino Zuljevic, in Advances in Computers, 2010

3.1.3.2 Real-Time Processing with Hybrid High-Performance Computing Hardware Architectures

We have implemented the design for the inverse Hadamard transform computation on a single node of a Cray XD-1. The full implementation of the design uses 22% of the slices (5201 out of 23,616) of the Virtex II-Pro XC2VP50 FPGA on the XD-1's node. It uses 36% of the BlockRAM elements (85 out of 232) and only 1% of the 18 × 18 multipliers (4 out of 232). The design also uses two of the four 4-MB QDR SRAM memory banks available on the XD-1 node. The design achieved a clock frequency of 142 MHz out of a maximum achievable frequency of 199 MHz. The maximum data transfer bandwidth on the XD-1 node is 1.42 GBps. However, this bandwidth can only be achieved under a design that operates at the maximum frequency of 199 MHz. Designs operating at lower clock frequencies achieve a proportional bandwidth. In our case, we achieved a data rate of 1.01 GBps.

Our data set consists of 100 frames of 520 scans each, with 3380 16-bit elements per scan. The total size of the data set is thus 400 MB. The streaming time at approximately 900 MBps is 0.46 s, which includes the ongoing frame accumulation processing. The inverse transform process for the accumulated frame takes 0.11 s and is twice as fast as executing the inverse transform algorithm on the host 2.4-GHz AMD Opteron CPU. The data write-back time for the results of the inverse transform is 4 ms.

Our research has determined that it is feasible to use FPGAs for processing mass spectrometry data at very high rates (GBps range). We have also shown that is possible to implement domain-specific signal processing algorithms (inverse Hadamard transform) to operate in conjunction with high-speed data capture.

> Read full chapter

Type I (Linear) Data Hiding

Husrev T. Sencar, ... Ali N. Akansu, in Data Hiding Fundamentals and Applications, 2004

to processing noise. Similarly, low-quality JPEG affects the high-frequency bands of subband decomposition (using an 8-tap binomial QMF [Daubechies] filter) to a much larger extent than the high-frequency Hadamard bands. We already know that low-frequency bands are not efficient channels due to the presence of high image noise. If the high-frequency bands are also affected by processing, it leaves us a small number of useful midfrequency bands. Transforms with lower GTC have many more of these useful midfrequency bands than the high-GTC transforms due to their spectral properties at higher processing noise scenarios. Therefore, *decompositions unsuitable for compression would, in general, be more immune to processing noise than decompositions with high GTC*. Also, recall that in Section 4.3, embedding in the image domain (or using identity transform for the transform blocks in Fig. 4-2) was found to be very robust to processing noise. The identity transform, which has the lowest GTC, has the highest robustness to processing noise. It is relevant to point out here that the term *robustness* is a measure of the change in *overall capacity* with a change in the processing noise (or processing scenario). The more robust the decomposition, the less is the reduction in capacity for a scenario of increased processing noise (or lower-quality compression). One should note that the robustness of the *low-frequency bands* of, say, the DCT decomposition will be much higher than the robustness of the single band coefficients (pixels) in the image domain. However, the low-frequency bands of the DCT have very little capacity due to high image noise. The reduced robustness of DCT is because of the drastic reduction in the *overall capacity* due to the significant increase of processing noise in the high-frequency bands.

to processing noise. Similarly, low-quality JPEG affects the high-frequency bands of subband decomposition (using an 8-tap binomial QMF [Daubechies] filter) to a much larger extent than the high-frequency Hadamard bands. We already know that low-frequency bands are not efficient channels due to the presence of high image noise. If the high-frequency bands are also affected by processing, it leaves us a small number of useful midfrequency bands. Transforms with lower GTC have many more of these useful midfrequency bands than the high-GTC transforms due to their spectral properties at higher processing noise scenarios. Therefore, *decompositions unsuitable for compression would, in general, be more immune to processing noise than decompositions with high GTC*. Also, recall that in Section 4.3, embedding in the image domain (or using identity transform for the transform blocks in Fig. 4-2) was found to be very robust to processing noise. The identity transform, which has the lowest GTC, has the highest robustness to processing noise. It is relevant to point out here that the term *robustness* is a measure of the change in *overall capacity* with a change in the processing noise (or processing scenario). The more robust the decomposition, the less is the reduction in capacity for a scenario of increased processing noise (or lower-quality compression). One should note that the robustness of the *low-frequency bands* of, say, the DCT decomposition will be much higher than the robustness of the single band coefficients (pixels) in the image domain. However, the low-frequency bands of the DCT have very little capacity due to high image noise. The reduced robustness of DCT is because of the drastic reduction in the *overall capacity* due to the significant increase of processing noise in the high-frequency bands.

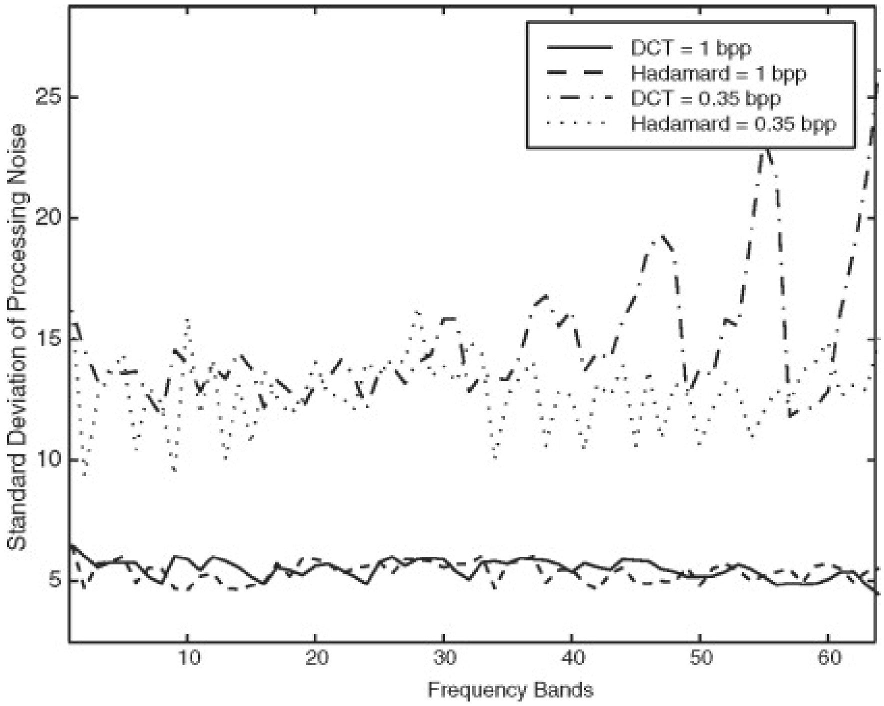


Figure 4-6. Comparison of standard deviation of processing noise for DCT and Hadamard decompositions. The source of processing noise is SPIHT compression at 1 bpp and 0.35 bpp.

Figure 4-6. Comparison of standard deviation of processing noise for DCT and Hadamard decompositions. The source of processing noise is SPIHT compression at 1 bpp and 0.35 bpp.

The next question that arises concerns the choice of the number of bands for the decomposition or size of the transform. From Eq. (4.15), we see that a decomposition will not hurt. At worst, it may cause no improvement. Therefore, decomposing each subchannel of, for instance, a 16-band decomposition further into four subchannels can only improve the capacity of data hiding, at least when processing noise is low.

> Read full chapter

Calibration Transfer, Part 2: The Instrumentation Aspects

Howard Mark, Jerry Workman Jr., in Chemometrics in Spectroscopy (Second Edition), 2018

Instrument Types

Spectrometers, or more appropriately, spectrophotometers, come in many design types. There are instruments based on the grating monochromator with mechanical drive, grating monochromator with encoder drive, the Michelson interferometer in various forms, dispersive gratings with array detectors, interference and linear variable filter instruments with single detectors, linear variable filters with array detection, MEMS-based Fabry-Perot interferometers, digital transform actuator technologies, acousto-optic tunable filters (AOTF), Hadamard transform spectrometers, laser diodes, tunable lasers, various interference filter types, multivariate optical computing, and others. Calibration transfer from one design type spectrometer to the same type and manufacturer is challenging enough, but transfer of sophisticated multifactor PLS models across instruments of different design types can be truly daunting. The requirement of precise spectral shapes across instruments requires more sophisticated transforms than just *X*-(wavelength or frequency) and *Y*-axes (photometric value) corrections. Multivariate calibrations are developed for many data dimensions and require delicate adjustments across this multidimensional space to fit measured data precisely. The data wavelength, data spacing, photometric response, photometric linearity, resolution, instrument lineshape and symmetry, and other parameters must be nearly identical for multivariate calibrations to yield equivalent predictive results across different spectrometers and different design types of spectrometers.

> Read full chapter

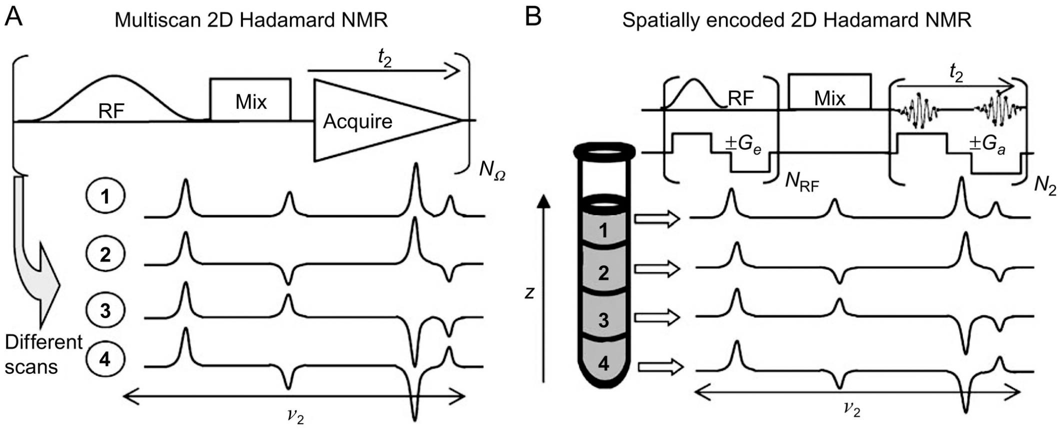
Ultrafast 2D NMR: Methods and Applications

Ultrafast 2D NMR: Methods and Applications

Boris Gouilleux, ... Patrick Giraudeau, in Annual Reports on NMR Spectroscopy, 2018

2.3.1.2 Hadamard

As for the spatial/spectral encoding method, only regions of interest are encoded in multiscan 2D Hadamard NMR. This is performed by modifying the phases of the different regions of interest through the different scans [23]. Thus, one encoded spectral region corresponds to a line of the Hadamard matrix. In multiscan 2D Hadamard NMR, traces are obtained thanks to FT in the direct dimension (Fig. 5A), which is followed by the application of Hadamard Transform (HT) in the indirect dimension, leading to the 2D spectrum.


A
Multiscan 2D Hadamard NMR
t_2
RF
Mix
Acquire
N_Ω
1
2
3
4
Different scans
ν_2
B
Spatially encoded 2D Hadamard NMR
RF
Mix
t_2
$\pm G_e$
N_{RF}
$\pm G_a$
N_2
z
1
2
3
4
ν_2

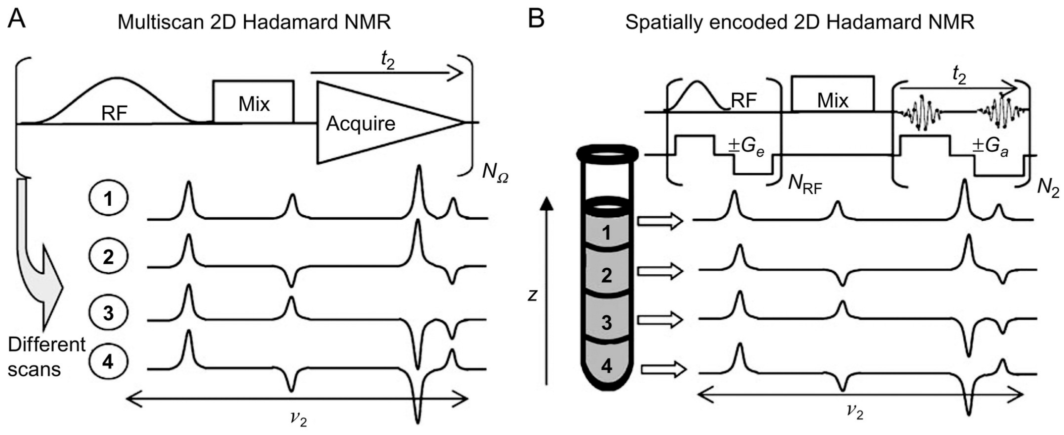


Fig. 5. (A) Original Hadamard-encoded scheme proposed for obtaining 2D NMR spectra. (B) UF 2D Hadamard approach based on a combined spectral/spatial encoding of the indirect-domain information. Both data are processed on the basis of a Hadamard transform which weights the spectra by a matrix—for the illustrated example—to afford the desired site-resolved information.

Fig. 5. (A) Original Hadamard-encoded scheme proposed for obtaining 2D NMR spectra. (B) UF 2D Hadamard approach based on a combined spectral/spatial encoding of the indirect-domain information. Both data are processed on the basis of a Hadamard transform which weights the spectra by a matrix—for the illustrated example—to afford the desired site-resolved information.

Reproduced from A. Tal, B. Shapira, L. Frydman, Single-scan 2D Hadamard NMR spectroscopy, Angew. Chem. Int. Ed. 48 (2009) 2732–2736 with permission from John Wiley and Sons.

Reproduced from A. Tal, B. Shapira, L. Frydman, Single-scan 2D Hadamard NMR spectroscopy, Angew. Chem. Int. Ed. 48 (2009) 2732–2736 with permission from John Wiley and Sons.

In ultrafast 2D Hadamard NMR, each line of the Hadamard matrix is spatially encoded, thanks to the use of spatio-spectral pulses. During the EPSI detection block, spatially resolved traces are obtained, as series of echoes, without FT (Fig. 5B). As in its multiscan counterpart, HT is applied in the ultrafast dimension to retrieve the 2D spectrum.

The use of Hadamard spatial encoding is particularly interesting when the ultrafast domain is sparse, as the gain in sensitivity is inversely proportional to the size of the Hadamard matrix (correlated to the number of spectral regions to be encoded). This technique is thus suitable for crowded spectra and, as for the spatial/spectral encoded experiments, requires a priori knowledge of the 1D spectrum.

> Read full chapter

Advanced Implementations

Husrev T. Sencar, ... Ali N. Akansu, in Data Hiding Fundamentals and Applications, 2004

6.2.7 Implementation and Simulation Results

Optimum codeword selection in multiple codebook hiding depends on designing the set of transform bases properly (i.e., they should be able to generate maximally separated transformations of the host signal; see Eq. (6.13)). One intuitive way of picking such a set of transform bases is by choosing them as rotation matrices so that each transformation of the host signal is a rotated version of the others. The multiple codebook data hiding method is implemented by designing the transformation bases using Givens rotations [66]. Givens rotations provide orthogonal transformations that can be employed to rotate a given vector with a chosen angle.

A particular transform basis is obtained by the consecutive multiplication of number of orthogonal matrices, all with determinant 1 so that the resulting is unitary. Each orthogonal matrix is derived from the identity matrix by introducing cos □k terms at (i, i) and (j, j) locations and sin □k and −sin □k terms at (i, j) and (j, i) locations in order to rotate the (i, j) coordinate plane with the designated angle □k. The rotation angles □k, k = 1, ..., L, are chosen by uniformly sampling .

By setting the signal size to N and the number of messages to M, the size of the codebooks utilized by the embedder is fixed to $M \times N$. The watermark signals that are embedded into the host signal are generated using a Hadamard transform matrix due to its simplicity. The Hadamard transform matrix of size $N \times N$ and its negated version are combined into a $2N \times N$ binary valued matrix. Every row of the combined matrix is indexed from 1 to $M = 2N$, scaled by for maximum separation, and assigned to the watermark signal $\mathbf{W}j$, $1 \leq j \leq M$, such that and $i \neq j + N$. The host signal and channel noise are *iid* zero-mean Gaussian vectors with . Prior to embedding, the permitted embedding distortion P_E is fixed, and the optimal values for the embedding parameter Δ are derived for the considered WNRs. The Δ values are also revealed to the detector. The parameters ▯ and ▯, however, are properly adjusted for each embedding in order to ensure an embedding distortion of P_E and are not known to the detector. The simulations are done for a different number of transformations L and signal sizes N by embedding and detecting randomly chosen message indices.

By setting the signal size to N and the number of messages to M, the size of the codebooks utilized by the embedder is fixed to $M \times N$. The watermark signals that are embedded into the host signal are generated using a Hadamard transform matrix due to its simplicity. The Hadamard transform matrix of size $N \times N$ and its negated version are combined into a $2N \times N$ binary valued matrix. Every row of the combined matrix is indexed from 1 to $M = 2N$, scaled by for maximum separation, and assigned to the watermark signal $\mathbf{W}j$, $1 \leq j \leq M$, such that and $i \neq j + N$. The host signal and channel noise are *iid* zero-mean Gaussian vectors with . Prior to embedding, the permitted embedding distortion P_E is fixed, and the optimal values for the embedding parameter Δ are derived for the considered WNRs. The Δ values are also revealed to the detector. The parameters ▯ and ▯, however, are properly adjusted for each embedding in order to ensure an embedding distortion of P_E and are not known to the detector. The simulations are done for a different number of transformations L and signal sizes N by embedding and detecting randomly chosen message indices.

Multiple codebook hiding is implemented in the type II scheme based on thresholding and distortion-compensation types of postprocessing using both maximum correlation and minimum distance criteria. Message embedding and detection with up to 25 codebooks is performed considering codebook sizes of 64 × 32, 128 × 64, and 256 × 128 and the WNR range of 0.1 to 1. Figures 6-23 and 6-24 display the probability of success results obtained, respectively, for $L = 1, 3$ and $L = 1, 4$ with varying N values when the postprocessing is thresholding. The increase in the embedding signal size N, at a fixed number of codebooks, improves the detection statistics because normalized correlation and mean squared distance give more reliable results with the larger signal sizes. On the other hand, Figs. 6-25 and 6-26 display the performance for a thresholding type of postprocessing when $N = 128$ and $L = 1, 3, 5, 9, 14, 25$ using the two criteria. Corresponding results for the distortion-compensation type of postprocessing are displayed in Figs. 6-27 and 6-28 for both criteria. It is observed from these performance simulations that the multiple codebook data hiding method has superior performance vs the corresponding single codebook method at the same N.

Multiple codebook hiding is implemented in the type II scheme based on thresholding and distortion-compensation types of postprocessing using both maximum correlation and minimum distance criteria. Message embedding and detection with up to 25 codebooks is performed considering codebook sizes of 64 × 32, 128 × 64, and 256 × 128 and the WNR range of 0.1 to 1. Figures 6-23 and 6-24 display the probability of success results obtained, respectively, for $L = 1, 3$ and $L = 1, 4$ with varying N values when the postprocessing is thresholding. The increase in the embedding signal size N, at a fixed number of codebooks, improves the detection statistics because normalized correlation and mean squared distance give more reliable results with the larger signal sizes. On the other hand, Figs. 6-25 and 6-26 display the performance for a thresholding type of postprocessing when $N = 128$ and $L = 1, 3, 5, 9, 14, 25$ using the two criteria. Corresponding results for the distortion-compensation type of postprocessing are displayed in Figs. 6-27 and 6-28 for both criteria. It is observed from these performance simulations that the multiple codebook data hiding method has superior performance vs the corresponding single codebook method at the same N.

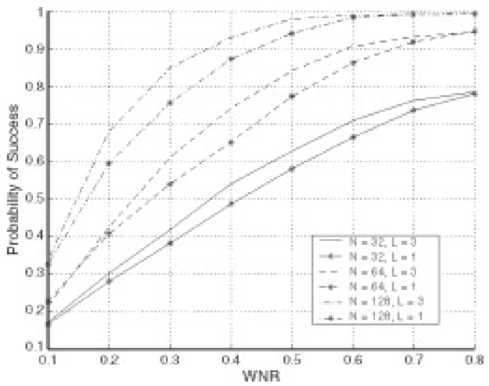


Figure 6-23. Probability of success performance for three-codebook hiding based on thresholding processing and maximum correlation criterion for various watermark signal sizes of *N* =32, *N* = 64, and *N* =128.

Figure 6-23. Probability of success performance for three-codebook hiding based on thresholding processing and maximum correlation criterion for various watermark signal sizes of *N* =32, *N* = 64, and *N* =128.

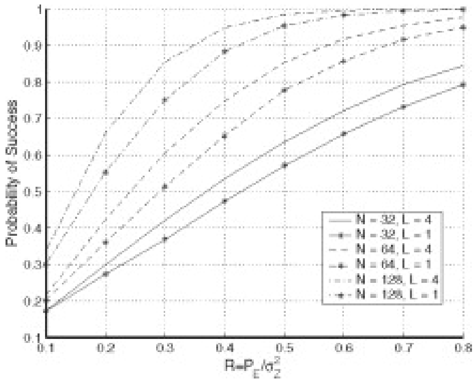


Figure 6-24. Probability of success performance for four-codebook hiding based on thresholding processing and minimum distance criterion for various watermark signal sizes of *N* =32, *N* =64, and *N* =128.

Figure 6-24. Probability of success performance for four-codebook hiding based on thresholding processing and minimum distance criterion for various watermark signal sizes of *N* =32, *N* =64, and *N* =128.

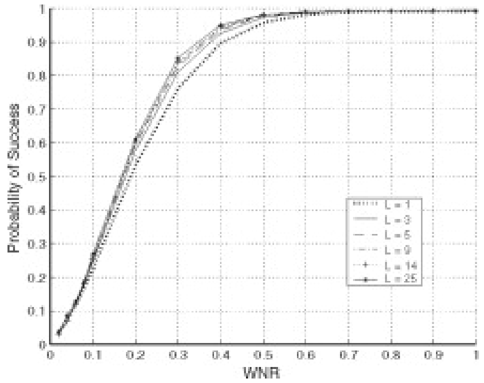


Figure 6-25. Probability of success performance for multiple codebook hiding based on thresholding type of processing and maximum correlation criterion for *L* = 1,3,5,9,14,25 and *N* =128.

Figure 6-25. Probability of success performance for multiple codebook hiding based on thresholding type of processing and maximum correlation criterion for *L* = 1,3,5,9,14,25 and *N* =128.

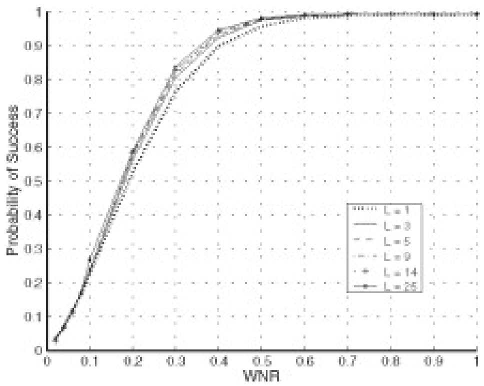
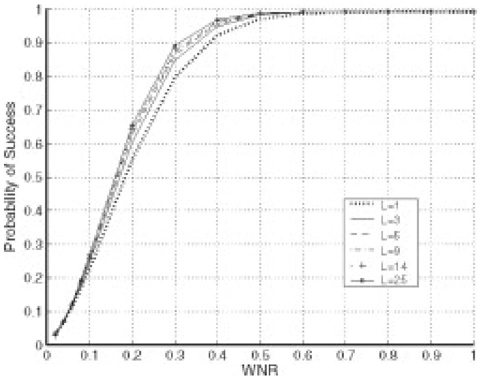


Figure 6-26. Probability of success performance for multiple codebook hiding based on thresholding type of processing and minimum distance criterion for *L* = 1,3,5,9,14,25 and *N* =128.

Figure 6-26. Probability of success performance for multiple codebook hiding based on thresholding type of processing and minimum distance criterion for *L* = 1,3,5,9,14,25 and *N* =128.


Probability of Success
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
WNR
L=1
L=3
L=5
L=9
L=14
L=25

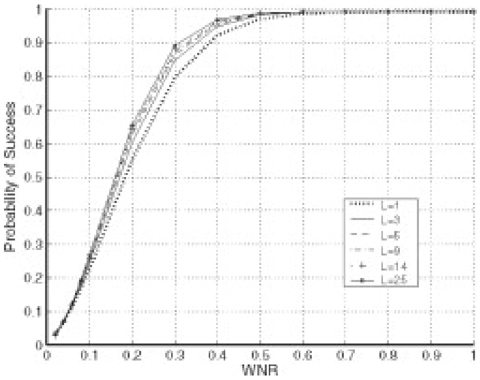


Figure 6-27. Probability of success performance for multiple codebook hiding based on distortion-compensation type of processing and maximum correlation criterion for $L = 1,3,5,9,14,25$ and $N = 128$.

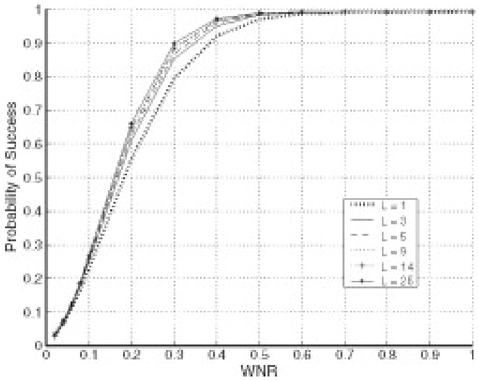


Figure 6-28. Probability of success performance for multiple codebook hiding based on distortion-compensation type of processing using minimum distance criterion for $L = 1,3,5,9,14,25$ and $N = 128$.

The computational complexity of the proposed method depends on the number of codebooks employed. Multiple codebook embedding, when compared with single codebook embedding, requires the embedding of the watermark signal into transformations of the host signal and comparison based on the resulting signals in order to select the transform basis. On the other hand, at the detector, extraction should be repeated for each transformation basis. Therefore, the computational complexity increases almost linearly with the number of codebooks (see Fig. 6-10).

> Read full chapter

Video Coding Standards

David R. Bull, in Communicating Pictures, 2014

12.7.5 The video coding layer

Intra-coding

standard also includes an transform as this is often better suited to content with higher spatial resolutions.

standard also includes an transform as this is often better suited to content with higher spatial resolutions.

Multiple reference frames: Up to 16 frames (or 32 fields) (depending on profile) can be buffered in the H.264 reference buffer. The reference buffer does not have to contain the 16 most recent frames, frames can be updated according to their utility, provided that the encoder and decoder remain synchronized. For further details on *MRF-ME*, the reader is referred back to Chapter 9.

Prediction structures: The H.264 coding tree enables a much more flexible approach to temporal prediction. For example, in the case of low delay, low memory constraints, an structure is most suitable, as offered by the baseline profile. If some delay is tolerable, along with the availability of B-frames, and multiple reference frames, then improved compression performance can be achieved. An example of this is shown in Figure 12.16, where a hierarchical prediction structure is imposed. Using this approach it has been demonstrated [12] that, if the quantization parameter is increased in a controlled fashion, with layer, then more efficient compression performance can be achieved.-

Figure 12.16. Hierarchical B-pictures in H.264.

Weighted prediction: This is a useful tool, introduced in H.264, that enables the weighting/offsetting of prediction samples values in B-frames. This provides coding gains in the case of accelerating motion or during scene cuts.

Quarter-pixel prediction: H.264 benefits from the availability of subpixel motion compensation to quarter-pixel accuracy. This has been estimated to provide up to 20% saving over the case of integer-pixel accuracy. A comprehensive description of the H.264/AVC sub-pixel motion estimation and compensation process was provided in Chapter 9. H.264/AVC uses a five-tap filter for half-pixel interpolation and a simple two-tap filter for local refinement to quarter-pixel accuracy.

Deblocking operations

The H.264 deblocking filter has been assessed to reduce the coded bit rate for the same subjective quality by up to 10%. It improves subjective visual and objective quality of the decoded picture. This highly content-adaptive non-linear filter removes blocking artifacts and does not unnecessarily blur the visual content. Its structure and operation were described in Chapter 9.

Variable length coding

- **Exp-Golomb coding:** This is a highly structured and simple variable length coding method, used almost universally in H.264 for all symbols apart from transform coefficients.

-

- **CAVLC:** Context-Adaptive VLC is a relatively low complexity but effective method, used as the Baseline and Extended Profiles for entropy coding of transform coefficients. Local contexts are used at the encoder to select between different VLC tables according to the local statistics, in this case the number of non-zero coefficients in neighboring blocks.
- **CABAC:** CABAC is a more complex, arithmetic coding-based method that is only supported in the H.264 Main and High Profiles. It uses local contexts to adjust its conditioning probabilities. CABAC has been found to reduce bit rate by between 10 and 20% compared to CAVLC, dependent on content type and quantization level.

Further details on all the above entropy coding methods can be found in Chapter 7.

Coder control

Coder control is a non-normative part of H.264/AVC. It is however a key element in achieving the optimum performance from the standard, since without good RDO, many of the advanced features (e.g. block size selection) included will not deliver major savings. The goal of coder control is to select what parts of the video signal should be encoded using which methods and with what parameter settings. Often a Lagrangian optimization approach is adopted where the distortion measure, , is based on sum of squared difference and the rate, , includes all bits associated with the decision including header, motion, side information, and transform data. Further details on the RDO methods used in H.264/AVC can be found in Chapter 10.

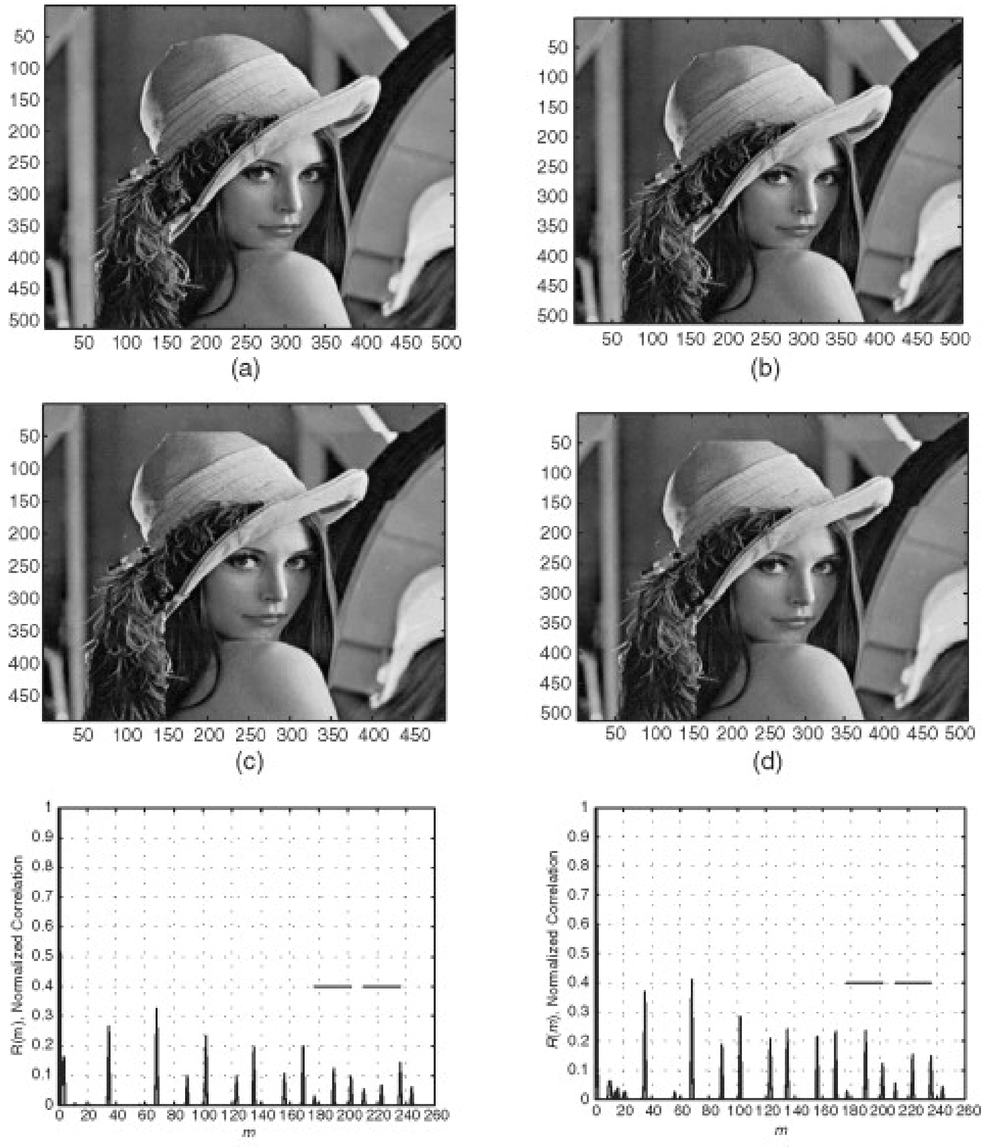
> Read full chapter

Major Design Issues

Husrev T. Sencar, ... Ali N. Akansu, in Data Hiding Fundamentals and Applications, 2004

7.2.4 Results

We implemented the methodology on a 512 × 512 gray level Lena image, Fig. 7-3-a. Message *m* is assumed to be a sequence of 32 bits. The signal W takes the form of the watermark signal corresponding to *m* with constraint on the correlation properties. Hadamard transform matrix is designed as the codebook and its orthogonal rows are mapped to codewords that are employed in watermark signal generation.


50
100
150
200
250
300
350
400
450
500
R(m), Normalized Correlation
m
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
20
40
60
80
100
120
140
160
180
200
220
240
260

(a)

(b)

(c)

(d)

(e)

(f)

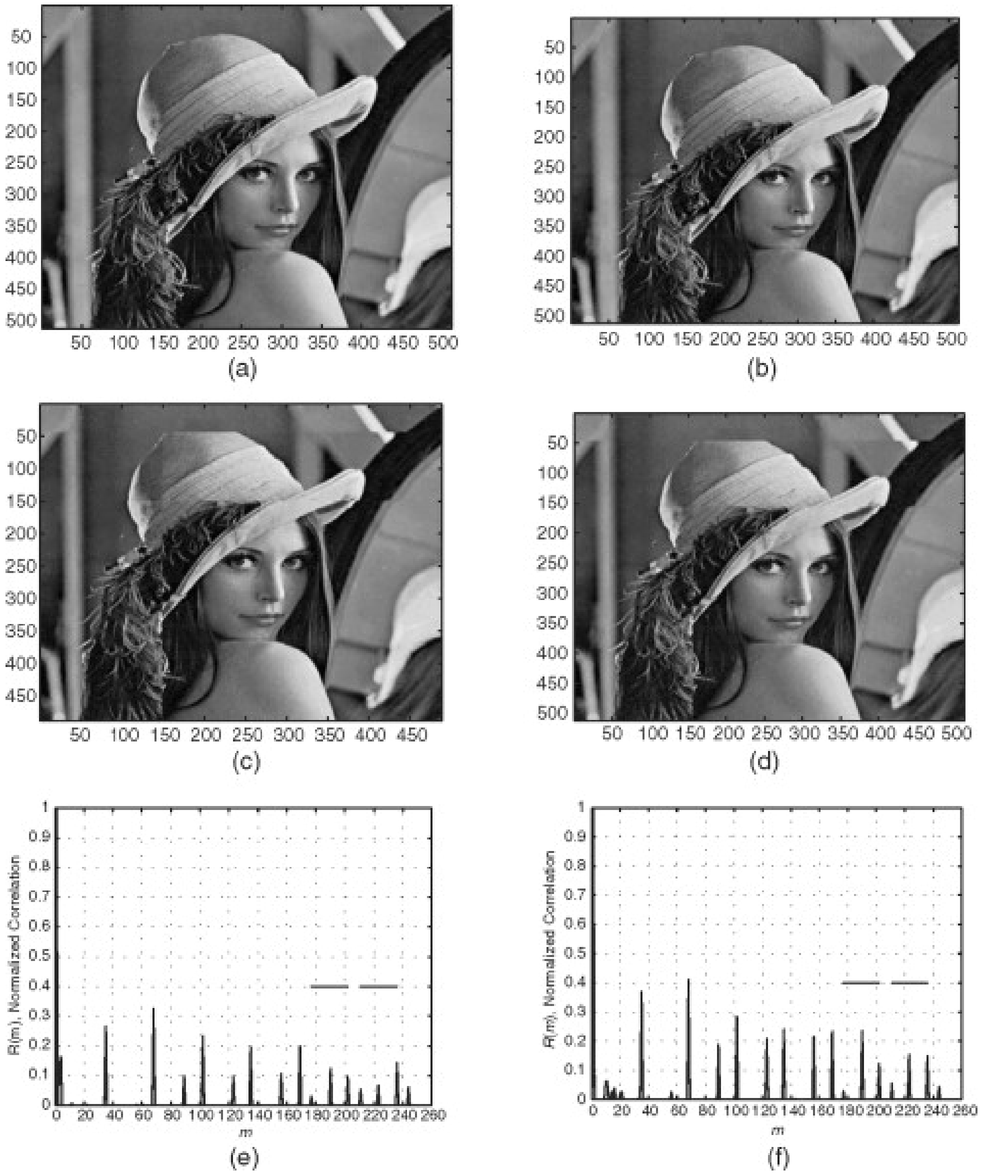


Figure 7-3. (a) Lena image. (b) Watermarked image. (c) Cropped image after watermarking. (d) Resampled image after cropping, and estimation of cropped amounts from the resampled image by projecting the two-dimensional autocorrelation function onto (e) horizontal dimension and (f) vertical dimension.

The message bit sequence is parsed into words. Then the message words are redundancy coded using RS error correcting codes. Using the codebook, encoded message words are BPSK (binary phase shift keying) modulated and ordered in a way that fulfills the frequency symmetry requirements for the phase of the all-pass filter in order to generate the watermark signal **W**. The watermark signal is chosen to be a 32 × 32 all-pass filter, which provides the hider with phase samples to modulate by the coded message *m*. Then, 16 copies of the watermark signal are embedded throughout the whole image.

The watermarked image is cropped, and in order to compensate the reduction in size, it is resampled back to its original size. At the extractor, a copy of the watermarked, cropped, and resampled image is divided into partitions of size **W**. Watermark detection for each partition is followed by the two-dimensional cyclic autocorrelation of the detected set of signals. Using correlation peak pattern, resampling factor ▯ is estimated. The extractor, knowing an estimate of the total cropped amount but not their locations, resamples the image back to its size after cropping. Hence, the disturbing effects of the resampling can be reversed or at least minimized. This image is then repartitioned and the watermark extracted. Since extracted watermark signals may have been cropped and translated, an immediate detection of message *m* is not possible. RS codes are used to detect the message *m* from the extracted watermark signal, since they are capable of correcting burst error. The two-dimensional signal is shifted in rows and columns until an errorless decoding is possible. High redundancy coding helps in detecting message *m* even under severe signal loss.

autocorrelation of the detected set of signals. Using correlation peak pattern, resampling factor □ is estimated. The extractor, knowing an estimate of the total cropped amount but not their locations, resamples the image back to its size after cropping. Hence, the disturbing effects of the resampling can be reversed or at least minimized. This image is then repartitioned and the watermark extracted. Since extracted watermark signals may have been cropped and translated, an immediate detection of message *m* is not possible. RS codes are used to detect the message *m* from the extracted watermark signal, since they are capable of correcting burst error. The two-dimensional signal is shifted in rows and columns until an errorless decoding is possible. High redundancy coding helps in detecting message *m* even under severe signal loss.

Figure 7-3a–d displays the results for the described method applied on the Lena image, Fig. 7-3a. The watermarked image is displayed in Fig. 7-3b, where the MSE per coefficient due to embedding is 6.9 (40 dB in PSNR). Figure 7-3c is the watermarked image cropped twice in both dimensions to a size of 488 × 488. Each cropping is the erasure of 12 lines of pixels in either horizontal or vertical dimension. The cropped image is resampled back to its original size of 512 × 512 in 7-3d. Figure 7-3e–f shows the projections of the cyclic autocorrelation function onto horizontal and vertical dimensions. The distance between the first and last peaks in each period, corresponding to the size of the watermark signal altered by the resampling factor, of the cyclic autocorrelation function is $d = 25$, which has an estimation error of 1 line of pixels in both dimensions. T_c is also measured using the Fig. 7-3e–f as 33 at some shifts and as 34 at most of the others. Then the image in Fig. 7-3d is resampled to a size shorter by 24 lines of pixels in each dimension, partitioned in 32 × 32 blocks and watermark detected. Extracted signals from each block are averaged. Then the averaged signal block is decoded in cyclic shifts of rows and columns until an errorless decoding is possible. For the presented implementation, the redundancy rate is around. RS codes were successful in detecting the 32-bit message *m* with no errors.

> Read full chapter

Fast DCT/DST Algorithms

Vladimir Britanak, ... K.R. Rao, in Discrete Cosine and Sine Transforms, 2007

4.4.3.2 DCT-II computation via Walsh–Hadamard transform

Since any EOT (see Section 4.3) can be expressed in terms of any other EOT through a conversion matrix [34], the DCT-II with its even symmetry/even antisymmetry structure of basis vectors can be realized via other simpler EOT [4, 28, 34, 37] such as sequency (Walsh) ordered Walsh–Hadamard transform (WHT) [1] whose basis vectors consist of ±1 elements only.

of basis vectors can be realized via other simpler EOT [4, 28, 34, 37] such as sequency (Walsh) ordered Walsh–Hadamard transform (WHT) [1] whose basis vectors consist of ±1 elements only.

Denote the DCT-II and WHT respectively in the matrix–vector notation (the normalization factors are omitted) as

(4.58)

Examination of and W_N matrices for a given N shows that there is a one-to-one correspondence between the sequencies (the number of zero-crossings in sign) of the rows of these transform matrices. This implies that the even symmetry/even antisymmetry structure of the WHT matrix is preserved in the DCT-II matrix. If we rearrange the rows of and W_N in bit-reversed order, then (4.58) can be rewritten as

(4.59)

Since is an orthonormal matrix, substituting it into the relation (4.59) we get [34]

(4.60)

where T_N is the conversion matrix which takes the Walsh domain vector and converts it to the DCT-II domain. The conversion matrix T_N has two important properties:

1. It is orthonormal, being the product of two orthonormal matrices and
2. It has a sparse block diagonal structure as long as the rows of and are in bit-reversed order.

Equation (4.60) defines the fast algorithm for DCT-II computation via WHT involving two steps: the efficient WHT computation and the implementation of the conversion matrix T_N. In order to illustrate the DCT-II computation via WHT consider $N =$ 8. The W_8 matrix is given by [1]

and the conversion matrix in analytical form is given by

where

The $M \times M$ block matrices U_M in the conversion matrix T_N possessing the following general structure [38]

(4.61)

can be further factorized into a product of sparse matrices. The higher-order conversion matrix T_{2N} can be generated recursively [37], i.e., having derived the block matrices for T_N we need to derive for the T_{2N} only the block matrix U_N taking into account its above general structure.

matrices for T_N we need to derive for the $T_{2}N$ only the block matrix U_N taking into account its above general structure.

Among the existing sparse factorizations of U_4 matrix [35, 37], the factorization presented in Ref. [37] is preferred in terms of structural simplicity and regularity and it is defined as

Among the existing sparse factorizations of U_4 matrix [35, 37], the factorization presented in Ref. [37] is preferred in terms of structural simplicity and regularity and it is defined as

where

where

For the fast WHT computation, Manz's algorithm is presented in Ref. [1] requiring the input data sequence $\{x_n\}$ to be in bit-reversed order. Since WHT is an EOT according to (4.17) we can obtain more suitable factorization of matrix as

For the fast WHT computation, Manz's algorithm is presented in Ref. [1] requiring the input data sequence $\{x_n\}$ to be in bit-reversed order. Since WHT is an EOT according to (4.17) we can obtain more suitable factorization of matrix as

where

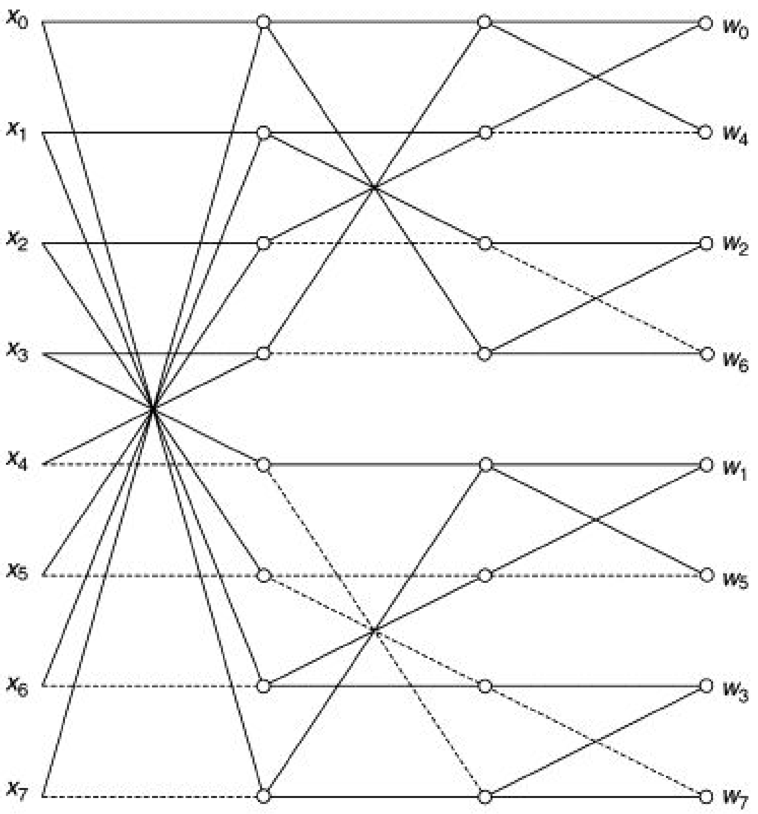
where

and

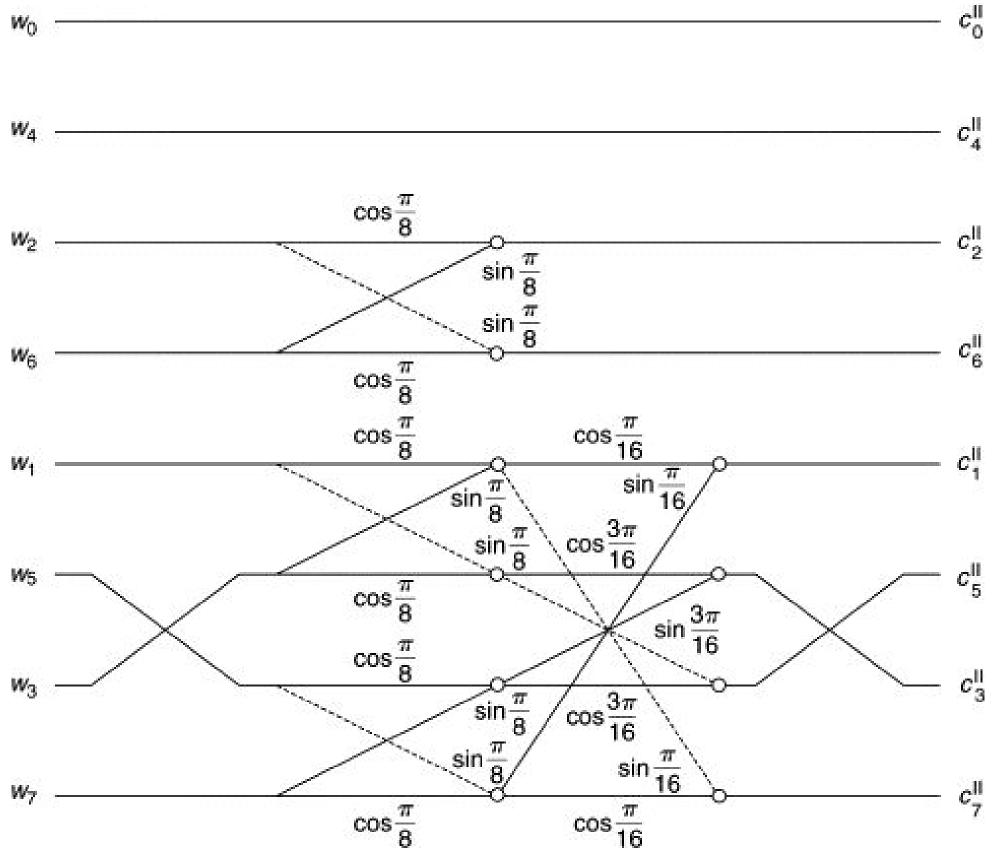
and

defining the new alternative fast WHT algorithm for $N = 8$. The corresponding signal flow graph for the DCT-II computation via WHT for $N = 8$ including $N = 4$ is shown in Fig. 4.12.

defining the new alternative fast WHT algorithm for $N = 8$. The corresponding signal flow graph for the DCT-II computation via WHT for $N = 8$ including $N = 4$ is shown in Fig. 4.12.



x_0
x_1
x_2
x_3
x_4
x_5
x_6
x_7
w_0
w_4
w_2
w_6
w_1
w_5
w_3
w_7



w_0
w_4
w_2
w_6
w_1
w_5
w_3
w_7
c_0^{II}
c_4^{II}
c_2^{II}
c_6^{II}
c_1^{II}
c_5^{II}
c_3^{II}
c_7^{II}
$\cos\frac{\pi}{8}$
$\sin\frac{\pi}{8}$
$\sin\frac{\pi}{8}$
$\cos\frac{\pi}{8}$
$\cos\frac{\pi}{8}$
$\cos\frac{\pi}{16}$
$\sin\frac{\pi}{8}$
$\sin\frac{\pi}{16}$
$\sin\frac{\pi}{8}$
$\cos\frac{3\pi}{16}$
$\cos\frac{\pi}{8}$
$\sin\frac{3\pi}{16}$
$\cos\frac{\pi}{8}$
$\sin\frac{\pi}{8}$
$\cos\frac{3\pi}{16}$
$\sin\frac{\pi}{8}$
$\sin\frac{\pi}{16}$
$\cos\frac{\pi}{8}$
$\cos\frac{\pi}{16}$

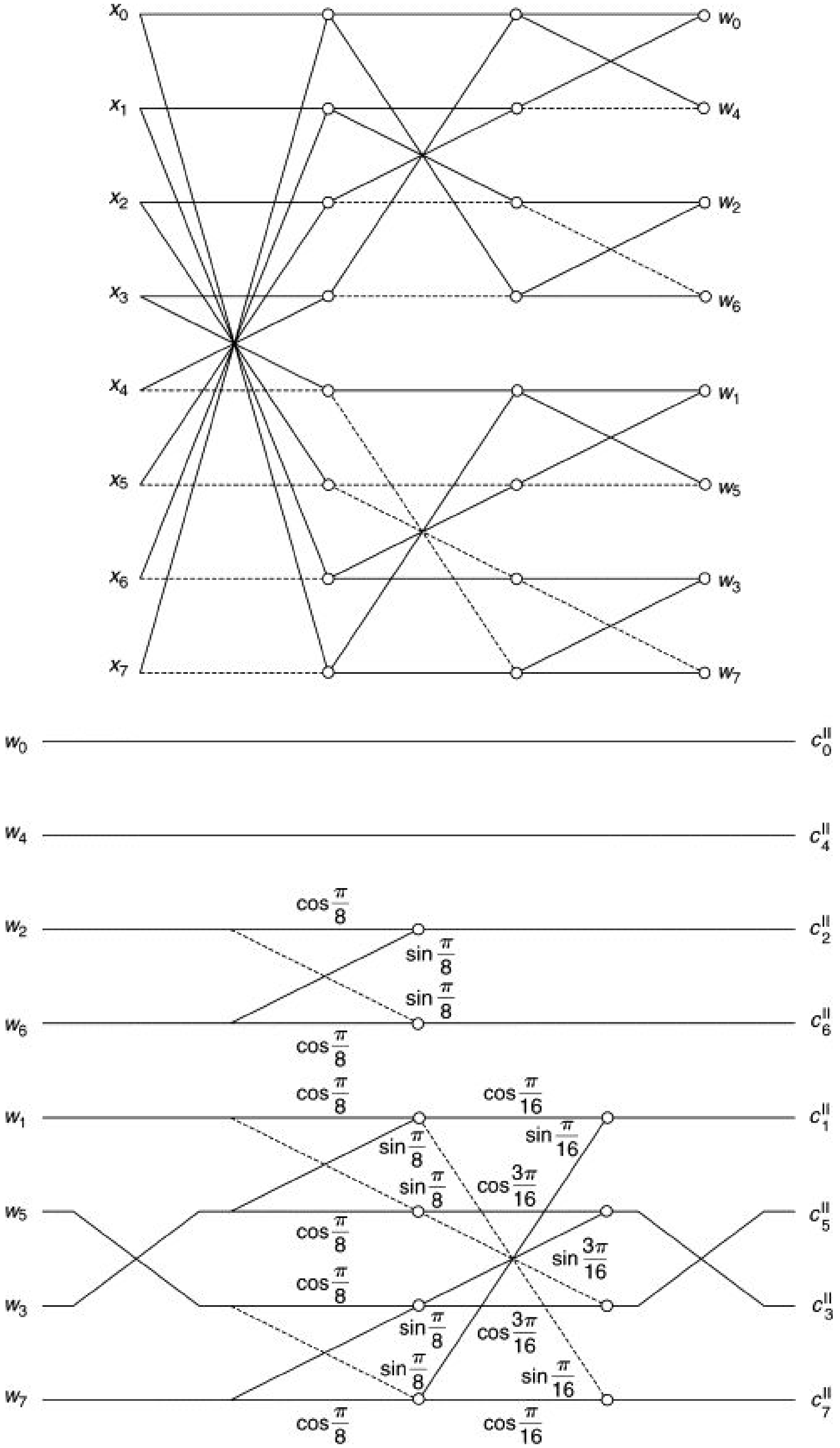


Fig. 4.12. The signal flow graph for the DCT-II computation via WHT for $N = 8$ (the fast WHT and conversion matrix).

Fig. 4.12. The signal flow graph for the DCT-II computation via WHT for $N = 8$ (the fast WHT and conversion matrix).

If we compare the signal flow graph for $N = 8$ in Fig. 4.12 with the DCT-II implementation via WHT presented in Ref. [35] the implementation here has the following advantages:

If we compare the signal flow graph for $N = 8$ in Fig. 4.12 with the DCT-II implementation via WHT presented in Ref. [35] the implementation here has the following advantages:

Saves two bit-reversal permutations.

- For 5 plane rotations the number of rotation angles is reduced from four to three.
- The implementation is more regular.

> Read full chapter

- Saves two bit-reversal permutations.
- For 5 plane rotations the number of rotation angles is reduced from four to three.
- The implementation is more regular.

> Read full chapter