If only the \( L \) values \( R_0 \) through \( R_{L-1} \) are required, then it can be seen by comparing (15) and (11) that the new algorithm will require fewer multiplications than the FFT method if
\[
L < 11.2 (1 + \log_2 N)
\] (16a)
and will require fewer additions if
\[
L < 5.6 (1 + \log_2 N).
\] (16b)
Therefore, we conclude that the new algorithm will generally be more efficient than the FFT method if
\[
N < 128 \] (17a)
or
\[
L < 10(1 + \log_2 N). \] (17b)

**CONCLUSION**

A new algorithm for computing the correlation of a block of sampled data has been presented. It is a direct method which trades an increased number of additions for a decreased number of multiplications. For applications where the "cost" (e.g., the time) of a multiplication is greater than that of an addition, the new algorithm is always more computationally efficient than direct evaluation of the correlation, and it is generally more efficient than FFT methods for processing 128 or fewer data points, or for calculating only the first \( L \) "lags" for \( L < 10 \log_2 2N \).

**REFERENCES**


**Discrete Cosine Transform**

N. AHMED, T. NATARAJAN, AND K. R. RAO

**Abstract**—A discrete cosine transform (DCT) is defined and an algorithm to compute it using the fast Fourier transform is developed. It is shown that the discrete cosine transform can be used in the area of digital processing for the purposes of pattern recognition and Wiener filtering. Its performance is compared with that of a class of orthogonal transforms and is found to compare closely to that of the Karhunen-Loève transform, which is known to be optimal. The performances of the Karhunen-Loève and discrete cosine transforms are also found to compare closely with respect to the rate-distortion criterion.

**Index Terms**—Discrete cosine transform, discrete Fourier transform, feature selection, Haar transform, Karhunen-Loève transform, rate distortion, Walsh-Hadamard transform, Wiener vector and scalar filtering.

**INTRODUCTION**

In recent years there has been an increasing interest with respect to using a class of orthogonal transforms in the general area of digital signal processing. This correspondence addresses itself towards two problems associated with image processing, namely, pattern recognition [1] and Wiener filtering [2].

In pattern recognition, orthogonal transforms enable a noninvertible transformation from the pattern space to a reduced dimensionality feature space. This allows a classification scheme to be implemented with substantially less features, with only a small increase in classification error.

In discrete Wiener filtering applications, the filter is represented by an \((M \times M)\) matrix \( G \). The estimate \( \hat{X} \) of data vector \( X \) is given by \( G \), where \( Z = X + N \) and \( N \) is the noise vector. This implies that approximately \( 2M^2 \) arithmetic operations are required to compute \( \hat{X} \). Use of orthogonal transforms yields a \( G \) in which a substantial number of elements are relatively small in magnitude, and hence can be set equal to zero. Thus a significant reduction in computation load is realized at the expense of a small increase in the mean-square estimation error.

The Walsh-Hadamard transform (WHT), discrete Fourier transform (DFT), the Haar transform (HT), and the slant transform (ST), have been considered for various applications [1], [2], [4]-[9] since these are orthogonal transforms that can be computed using fast algorithms. The performance of these transforms is generally compared with that of the Karhunen-Loève transform (KLT) which is known to be optimal with respect to the following performance measures: variation distribution [1], estimation using the mean-square error criterion [2], [4], and the rate-distortion function [5]. Although the KLT is optimal, there is no general algorithm that enables its fast computation [1].

In this correspondence, a discrete cosine transform (DCT) is introduced along with an algorithm that enables its fast computation. It is shown that the performance of the DCT compares more closely to that of the KLT relative to the performances of the DFT, WHT, and HT.

**Discrete Cosine Transform**

The DCT of a data sequence \( X(m), m = 0, 1, \cdots, (M - 1) \) is defined as
\[
G_x(0) = \frac{\sqrt{2}}{M} \sum_{m=0}^{M-1} X(m)
\]
\[
G_x(k) = \frac{2}{M} \sum_{m=0}^{M-1} X(m) \cos \left( \frac{(2m + 1) k \pi}{2M} \right), \quad k = 1, 2, \cdots, (M - 1)
\] (1)
where \( G_x(k) \) is the \( k \)th DCT coefficient. It is worthwhile noting that the set of basis vectors \( \{1/\sqrt{M}, \cos ((2m + 1) k \pi)/(2M) \} \) is actually a class of discrete Chebyshev polynomials. This can be seen by recalling that Chebyshev polynomials can be defined as [3]
\[
\hat{T}_0(\xi_p) = \frac{1}{\sqrt{2}}
\]
\[
\hat{T}_k(\xi_p) = \cos (k \cos^{-1} \xi_p), \quad k, p = 1, 2, \cdots, M
\] (2)
where \( \hat{T}_k(\xi_p) \) is the \( k \)th Chebyshev polynomial.

Now, in (2), \( \xi_p \) is chosen to be the \( p \)th zero of \( \hat{T}_M(\xi) \), which is given by [3]
\[
\xi_p = \cos \left( \frac{2(p - 1) \pi}{2M} \right), \quad p = 1, 2, \cdots, M.
\] (3)
Substituting (3) in (2), one obtains the set of Chebyshev polynomials
\[
\hat{T}_0(p) = \frac{1}{\sqrt{2}}
\]
\[
\hat{T}_k(p) = \cos \left( \frac{2(p - 1) k \pi}{2M} \right), \quad k, p = 1, 2, \cdots, M.
\] (4)
From (4) it follows that the \( \hat{T}_k(p) \) can equivalently be defined as
\[
T_0(m) = \frac{1}{\sqrt{2}}
\]
\[
T_k(m) = \cos \left( \frac{(2m + 1) k \pi}{2M} \right), \quad k = 1, 2, \cdots, (M - 1),
\]
m = 0, 1, \cdots, M - 1. (5)
Comparing (5) with (1) we conclude that the basis member \( \cos ((2m \pi + \pi) / M, \cdots, (M - 2) \pi / M) \) of the Chebyshev polynomials is the same as the set \( \{1/\sqrt{M}, \cos ((2m + 1) \pi / M) \} \) of the DCT.
1) \( \Pi/(2M) \) is the \( k \)th Chebyshev polynomial \( T_k(\xi) \) evaluated at the \( m \)th zero of \( T_M(\xi) \).

Again, the inverse cosine discrete transform (ICDT) is defined as

\[
X(m) = \frac{1}{\sqrt{2}} G_x(0) + \sum_{k=1}^{M-1} G_x(k) \cos \left( \frac{(2m+1)\Pi}{2M} \right),
\]

\[ m = 0, 1, \cdots, (M-1). \]  (6)

We note that applying the orthogonal property [3]

\[
\sum_{m=0}^{M-1} T_k(m) T_l(m) = \begin{cases}
\frac{M}{2}, & k = l = 0 \\
\frac{M}{2}, & k = l \neq 0 \\
0, & k \neq l
\end{cases}
\]  (7)

to (6) yields the DCT in (1).

If (6) is written in matrix form and \( \Lambda \) is the \((M \times M)\) matrix that denotes the cosine transformation, then the orthogonal property can be expressed as

\[
\Lambda^T \Lambda = \frac{M}{2} [I]
\]  (8)

where \( \Lambda^T \) is the transpose of \( \Lambda \) and \([I]\) is the \((M \times M)\) identity matrix.

**Motivation**

The motivation for defining the DCT is that it can be demonstrated that its basis set provides a good approximation to the eigenvectors of the class of Toeplitz matrices defined as

\[
\psi = \begin{bmatrix}
1 & \rho & \rho^2 & \cdots & \rho^{M-1} \\
\rho & 1 & \rho^2 & \cdots & \rho^{M-2} \\
0 & 0 & \cdots & \rho \\
\rho^{M-1} & \rho^{M-2} & \cdots & 1
\end{bmatrix}, \quad 0 < \rho < 1.
\]  (9)

For the purposes of illustration, the eigenvectors of \( \psi \) for \( M = 8 \) and \( \rho = 0.9 \) are plotted (see Fig. 1) against

\[
\left\{ \frac{1}{\sqrt{2}} \cos \left( \frac{(2m+1)\Pi}{16} \right), \quad k = 1, 2, \cdots, 7, m = 0, 1, \cdots, 7 \right\}
\]  (10)

which constitute the basis set for the DCT. The close resemblance (aside from the 180\(^\circ\) phase shift) between the eigenvectors and the set defined in (10) is apparent.

**Algorithm**

It can be shown that (1) can be expressed as

\[
G_x(0) = \frac{\sqrt{2}}{M} \sum_{m=0}^{M-1} X(m)
\]

\[
G_x(k) = \frac{2}{M} \text{Re} \left\{ e^{-i2\Pi k/M} \sum_{m=0}^{M-1} X(m) w^{km} \right\}, \quad k = 1, 2, \cdots, (M - 1)
\]  (11)

where

\[
w = e^{-i\Pi/2}, \quad i = \sqrt{-1}
\]

\[
X(m) = 0, \quad m = M, (M + 1), \cdots, (2M - 1)
\]

and \( \text{Re} \left\{ \cdot \right\} \) implies the real part of the term enclosed. From (11) it follows that all the \( M \) DCT coefficients can be computed using a 2M-point fast Fourier transform (FFT). Since (1) and (6) are of the same form, FFT can also be used to compute the ICDT. Similarly, if a discrete sine transform were defined, then the \( \text{Re} \left\{ \cdot \right\} \) in (11) would be replaced by \( \text{Im} \left\{ \cdot \right\} \), which denotes the imaginary part of the term enclosed.

**Computational Results**

In image processing applications, \( \psi \) in (9) provides a useful model for the data covariance matrix corresponding to the rows and columns of an image matrix [6], [7]. The covariance matrix in the transform domain is denoted by \( \Psi \) and is given by

\[
\Psi = \Lambda \psi \Lambda^* T
\]  (12)

where \( \Lambda \) is the matrix representation of an orthogonal transformation and \( \Lambda^* \) is the complex conjugate of \( \Lambda \). From (12) it follows that \( \Psi \) can be computed as a two-dimensional transform of \( \psi \).

**Feature Selection**

A criterion for eliminating features (i.e., components of a transform vector), which are least useful for classification purposes, was developed by Andrews [1]. It states that features whose variances (i.e., main diagonal elements of \( \Psi \)) are relatively large should be retained. (Fig. 2 should be retained.) Fig. 2 shows the various variances ranked in decreasing order of magnitude. From the information in Fig. 2 it is apparent that relative to the set of orthogonal transforms shown, the DCT compares most closely to the KLT.

**Wiener Filtering**

The role of orthogonal transforms played in filtering applications is illustrated in Fig. 3 [2]. \( Z \) is an \((M \times 1)\) vector which is the sum of a
vector $X$ and a noise vector $N$. $X$ is considered to belong to a random process whose covariance matrix is given by $\psi$ which is defined in (9). The Wiener filter $G$ is in the form of an $(M \times M)$ matrix. $A$ and $A^{-1}$ represent an orthonormal transform and its inverse, respectively, while $\hat{X}$ denotes the estimate of $X$, using the mean-square error criterion.

We restrict our attention to the case when $G$ is constrained to be a diagonal matrix $Q$. This class of Wiener filters is referred to as scalar filters while the more general class (denoted by $G$) is referred to as vector filters. The additive noise (see Fig. 3) is considered to be white, zero mean, and uncorrelated with the data. If the mean-square estimation error due to scalar filtering is denoted by $e_Q$, then $e_Q$ can be expressed as [4]

$$e_Q = 1 - \frac{1}{M} \sum_{s=1}^{M} \frac{\psi_x^2(s, s)}{\psi_x(s, s) + \psi_y(s, s)}$$

(13)

where $\psi_x$ and $\psi_y$ denote the transform domain covariance matrices of the data and noise, respectively. Table I lists the values of $e_Q$ for different values of $M$ for the case $\rho = 0.9$ and a signal-to-noise ratio of unity. From Table I it is evident that the DCT comes closest to the KLT which is optimal. This information is presented in terms of a set of performance curves in Fig. 4.

**ADDITIONAL CONSIDERATIONS**

In conclusion, we compare the performance of the DCT with KLT, DFT, and the identity transforms, using the rate-distortion criterion [5]. This performance criterion provides a measure of the information rate $R$ that can be achieved while still maintaining a fixed distortion $D$, for encoding purposes. Considering Gaussian sources along with the mean-square error criterion, the rate-distortion performance measure is given by [5]

$$R(A, D) = \frac{1}{2M} \sum_{j=1}^{M} \max \left\{ 0, \ln \left( \frac{g_j}{\bar{g}} \right) \right\}$$

(14a)

![Fig. 3. Wiener filtering model.](image)

**Fig. 3.** Wiener filtering model.

**TABLE I**

<table>
<thead>
<tr>
<th>Transform</th>
<th>$M$ 2</th>
<th>$M$ 4</th>
<th>$M$ 8</th>
<th>$M$ 16</th>
<th>$M$ 32</th>
<th>$M$ 64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Karhunen-Loève</td>
<td>0.3730</td>
<td>0.2915</td>
<td>0.2533</td>
<td>0.2356</td>
<td>0.2268</td>
<td>0.2224</td>
</tr>
<tr>
<td>Discrete cosine</td>
<td>0.3730</td>
<td>0.2920</td>
<td>0.2546</td>
<td>0.2374</td>
<td>0.2282</td>
<td>0.2232</td>
</tr>
<tr>
<td>Discrete Fourier</td>
<td>0.3730</td>
<td>0.2964</td>
<td>0.2706</td>
<td>0.2592</td>
<td>0.2441</td>
<td>0.2320</td>
</tr>
<tr>
<td>Walsh-Hadamard</td>
<td>0.3730</td>
<td>0.2942</td>
<td>0.2649</td>
<td>0.2582</td>
<td>0.2582</td>
<td>0.2559</td>
</tr>
<tr>
<td>Haar</td>
<td>0.3730</td>
<td>0.2942</td>
<td>0.2650</td>
<td>0.2589</td>
<td>0.2581</td>
<td>0.2581</td>
</tr>
</tbody>
</table>

![Fig. 4. Mean-square error performance of various transforms for scalar Wiener filtering; $\rho = 0.9$.](image)

**Fig. 4.** Mean-square error performance of various transforms for scalar Wiener filtering; $\rho = 0.9$. 

Authorized licensed use limited to: IEEE Xplore. Downloaded on January 19, 2009 at 14:27 from IEEE Xplore. Restrictions apply.
Minimization of Linear Sequential Machines

CHI-TSONG CHEN

Abstract—An algorithm is presented to minimize linear sequential machines to reduced form.

Index Terms—Algorithm, linear machine, minimization, reduced machine, reduction, sequential machine.

I. INTRODUCTION

The minimization of a linear sequential machine to a reduced form is an important topic and is discussed in many texts [11]–[4]. The minimization procedure presented in [1]–[4] is as follows. Let \( \{A, B, C, D\} \) be an \( n \)-dimensional linear machine over \( GF(p) \), and let \( r \), with \( r < n \), be the rank of the diagnostic matrix \( K = \begin{bmatrix} C' & A' & \cdots & (A')^{n-1} & C' \end{bmatrix} \), where the prime stands for the transpose. Define an \( r \times n \) matrix \( T \) consisting of the first \( r \) linearly independent rows of \( K \), and an \( n \times r \) matrix \( R \) denoting the right inverse of \( T \) so that \( TR = I \). Then the linear machine \( \{TAR, TB, CR, D\} \) is a reduced form of \( \{A, B, C, D\} \). In this correspondence, an algorithm will be introduced to find a special set of \( r \) linearly independent rows in \( K \). A reduced machine can then be read out from this algorithm without the need of inverting any matrix. Furthermore, the reduced machine will be in a canonical form.

II. ALGORITHM

Let \( C \) be a \( q \times n \) matrix, and let \( c_i \) be the \( i \)th row of \( C \). The diagnostic matrix of \( \{A, B, C, D\} \) will be arranged in the following order:

\[
P = \begin{bmatrix}
c_1A \\
c_1A^2 \\
\vdots \\
c_1A^{r-1} \\
c_q \\
c_qA \\
\vdots \\
c_qA^{r-1}
\end{bmatrix}
\]

(1)

Now an algorithm will be introduced to find the first \( r \) linearly independent rows in \( P \). This is achieved by a series of elementary transformations. Let

\[
K_rK_{r-1} \cdots K_2K_1P = KP
\]

(2)

where \( K_i \) are lower triangular matrices with all diagonal elements unity, and are obtained in the following manner. Let \( p_i(j) \) be the first nonzero element from the left in the first row of \( P \), where \( j \) denotes the position. Then \( K_1 \) is chosen so that all, except the first two, elements of the \( j \)th column of \( K_1P \) are zero. Let \( p_2(j) \) be the first nonzero element from the left of the second row of \( K_1P \). Then \( K_2 \) is chosen so that all, except the first two, elements of the \( j \)th column of \( K_2K_1P \) are zero. Proceed in this manner until all linearly independent rows of \( P \) are found. Note that in this process, if one row is identically zero, then proceed to the next nonzero row. By multiplying these \( K_i \), we obtain

\[
K = K_rK_{r-1} \cdots K_2K_1.
\]

Note that \( KP \) will be finally of form

Acknowledgment

The authors wish to thank Dr. H. C. Andrews, University of Southern California, for his valuable suggestions pertaining to the rate-distortion computation.

REFERENCES