Comparative study of the discrete cosine transform and the discrete orthogonal Gauss-Hermite transform for biomedical signal compression

P. LAZARIDIS, G. DEBARGE*, P. GALLION*, Z. ZAHARIS, D. KAMPITAKI, A. HATZIGAIDAS, A. PAPASTERGIOU, G. GRAMMATIKOPOULOS**
Department of Electronics, (**) Department of Esthetics
Alexander Technological Educational Institute of Thessaloniki, EPEAEK II
Sindos, 57400 Thessaloniki, Macedonia, GREECE.
(*) Département Communications et Electronique, Unité de Recherche Associée au Centre National de la Recherche Scientifique, 820 Ecole Nationale Supérieure des Télécommunications, 46 rue Barrault, 75634 Paris Cedex 13, FRANCE.
pavloslazaridis@hotmail.com, debarge@enst.fr, gallion@enst.fr, zaharis@auth.gr, dimitra@her.forthnet.gr, sakis@el.teithe.gr, natp@el.teithe.gr

Abstract : An extensive comparison of the recently developed discrete orthogonal Gauss-Hermite transform (DOGHT) to the discrete cosine transform (DCT) used in many multimedia compression standards is presented in view of applications in the biomedical signal compression domain. The DOGHT is using a basis of orthonormal Gauss-Hermite functions in place of the cosine functions basis used by the DCT and it is shown to perform considerably better than the DCT in several cases, presenting similar reconstruction error for the double compression ratio. This study is limited in the comparison of very simple signals for the sake of clarity. The DOGHT combines some of the advantages of wavelet expansions with the simple analytic form and desirable mathematical properties of the DCT.

Key-Words: - Signal, Compression, Biomedical images, Discrete Orthogonal Gauss-Hermite Transform

1. Introduction

There is today a big demand for the compression of digital sound, image, video, biomedical image, etc. signals in order to facilitate their storage e.g. in CD-ROMs or hard disks of reasonable capacity but more importantly in order to facilitate their (wireless) transmission through limited bandwidth channels.

During the last 12 years many signal compression algorithms were proposed. The most widespread and normalised algorithms that constitute international standards are: JPEG (Joint Photographic Expert Group) for the compression of still picture since 1991 and MPEG (Moving Picture Expert Group) for the compression of moving pictures (video) and sound since 1992. More precisely, the simplest compression algorithm standard was presented in 1992 while the more complicated and enhanced subsequent versions are MPEG-2 (presented in 1994) and MPEG-4 (presented in 1996). For the sound compression the algorithm that is used is MPEG Layer 3 or MP3 that is included in the general MPEG video standard. Moreover, the JPEG algorithm is also widely used for the storage (e.g. in digital photographic cameras) and the transmission of still pictures through the Internet, GPRS, etc.

The basic ingredient and "heart" of all the above mentioned compression methods is the Discrete Cosine Transform – DCT, cf. [1], for which there exist since many years dedicated integrated circuits for its computation, cf. for example [2].

The MP3 and JPEG algorithms use respectively the DCT transform with N = 18 points and the 2-dimensional DCT with 8x8 = 64 points. These algorithms lead to a compression of about 10 times without any noticeable quality degradation.

Observations and experiments that were made during the last decade with the sound and picture compression standards MP3 and JPEG have shown that the distortion-free limit of '10 times compression' is mainly due to the DCT transform that is common in the two methods. Hence, in the quest for even higher compression ratios than the 1:10 and particularly wanting to achieve a less abrupt increase of picture distortion for the high compression ratios (e.g. 1:100) it was proposed to replace the DCT by the DWT (Discrete

Wavelet Transform). The latter is using wavelet basis functions in place of the cosine basis functions, cf. [3]. In this way, appeared the most recent standard for compression of still pictures known as JPEG2000 (.jp2) since 2000, cf. [4]. This compression algorithm achieves on average 30 % better compression than the JPEG algorithm, and this becomes visible especially at the higher compression ratios.

However, the need for even higher compression ratios in all the above mentioned applications remains and especially there is a need for relatively simple algorithms of reasonable computational complexity.

The DOGHT algorithm proposed by this paper can replace the above mentioned DCT and DWT transforms, shows better performance than both of them in most practical applications, while it is given in a simple analytical form similar to that of the DCT. Furthermore, for the DOGHT algorithm there exist mathematical properties (orthogonality, Parseval relation, exact inverse, etc.) very similar to those of the DCT and this facilitates considerably its implementation.

2. Description

The Discrete Orthogonal Gauss-Hermite Transform – DOGHT, [5-7], is defined by the relations:

$$c_n = \sum_{i=0}^{N-1} w_i f\left(\frac{t_i}{T}\right) h_n(t_i) \tag{1}$$

$$\tilde{f}\left(\frac{t_i}{T}\right) = \sum_{n=0}^{N-1} c_n \, h_n(t_i) \tag{2}$$

where T is a time scaling factor and

$$h_n(t) = \frac{1}{\left(2^n n! \sqrt{\pi}\right)^{1/2}} \exp\left(-\frac{t^2}{2}\right) H_n(t) \tag{3}$$

are the normalised Gauss-Hermite functions.

Furthermore,

$$\int_{-\infty}^{+\infty} h_m(t) h_n(t) \, dt = \delta_{mn} \tag{4}$$

is the continuous time orthogonality relation of the normalised Gauss-Hermite functions of order n : h_n and H_n are the classical Hermite polynomials of order n. The sampling points: $t_0 < t_1 < t_2 < K \, t_{N-1}$ are the zeroes of the Hermite polynomial of order N, well known from the Gauss quadrature approximate integration theory. The weights w_i are given by the relation:

$$w_i = \frac{2}{\left[h_N(t_i)\right]^2} \tag{5}$$

The normalised Gauss-Hermite functions satisfy the following very important discrete orthogonality relations (discrete time orthogonality, [8])

$$\sum_{i=0}^{N-1} w_i h_m(t_i) h_n(t_i) = \delta_{mn} \tag{6}$$

$$\sum_{n=0}^{N-1} w_j h_n(t_i) h_n(t_j) = \delta_{ij} \tag{7}$$

Because of equation (6) the transform coefficients c_n are calculated by equation (1). Moreover, equation (7) leads to the relation:

$$\tilde{f}\left(\frac{t_i}{T}\right) = f\left(\frac{t_i}{T}\right) \tag{8}$$

which is a basic and very important property of the DOGHT transform (collocation property). The above properties are very similar to the properties of the Discrete Cosine Transform – DCT and the result of these properties is the existence of an exact inverse DOGHT transform at the points *ti* of the Hermite sampling distribution.

In order to facilitate comparison it is mentioned that the Discrete Cosine Transform – DCT is defined by the relations:

$$c_n = \frac{2\alpha}{\mathrm{N}} \sum_{k=0}^{N-1} f(t_k) \cos\left[\frac{n\pi(k+1/2)}{N}\right] \tag{9}$$

$$\tilde{f}(t_k) = \sum_{n=0}^{N-1} \alpha \, c_n \cos\left[\frac{n\pi(k+1/2)}{N}\right] \tag{10}$$

$$\alpha = 1/\sqrt{2}, \, n = 0$$

$\alpha = 1, n > 0$

The sampling point distribution is uniform and is given in the symmetrical interval $[-t_{\max}, +t_{\max}]$ from the relation:

$$t_k = -t_{\max} + k\Delta t, \ \Delta t = \frac{2t_{\max}}{N-1}$$

The existence of an exact inverse is also a property of the DCT transform: $\tilde{f}(t_k) = f(t_k)$ at the points t_k of the uniform sampling.

The Parseval relation for the DOGHT transform is :

$$\tilde{E} = \frac{1}{T}\sum_{n=0}^{N-1} |c_n|^2 = \frac{1}{T}\sum_{i=0}^{N-1} w_i \left| f\left(\frac{t_i}{T}\right)\right|^2 \qquad (11)$$

where $\tilde{E}$ is the approximate signal energy. The exact signal energy is :

$$E = \int_{-\infty}^{+\infty} |f(t)|^2 dt \qquad (12)$$

The Parseval relation (11) is particularly useful: for the determination of the number of sampling points required for an accurate approximation of the signal energy, for the correct choice of the time scaling factor T, and in general for the estimation of the convergence rate of the transform.

The Parseval relation (11) reveals that in the case of the DOGHT transform the signal energy is approximated by a numerical integral (quadrature) of the Gauss-Hermite type while in the case of the DCT transform, as is well known from the DCT Parseval relation (13) the signal energy is approximated by a rectangle rule numerical integral :

$$\tilde{E} = \frac{N\Delta t}{2}\sum_{n=0}^{N-1} |c_n|^2 = \Delta t \sum_{k=0}^{N-1} |f(t_k)|^2 \qquad (13)$$

It is also well known that the numerical integral of the Gauss-Hermite type converges much more rapidly than the simple rectangle rule in the vast majority of cases.

Concerning the choice of the time scaling factor T of the DOGHT transform, and in analogy with the DCT transform, we first choose a computational time window $[-t_{\max}, t_{\max}]$ according to the time duration of the processed signal. For the time scaling factor the following relation is proposed :

$$T = \frac{t_{N-1}}{t_{\max}} \qquad (14)$$

where t_{N-1} is the biggest zero of the Hermite polynomial according to the Gauss-Hermite quadrature theory. Thus, with this choice, the first sampling point : $\frac{t_0}{T} = -t_{\max}$ and the last sampling point : $\frac{t_{N-1}}{T} = +t_{\max}$ coincide with the borders of the computational time window $[-t_{\max}, +t_{\max}]$.

Example 1

As an example we will calculate the DOGHT transform of the function $f(t) = \text{sec}h(t)$ in the time window $t \in [-5, +5]$ using N = 8 points. The results are presented in Table 1. The inversely transformed function in the time domain is given by:

$$\tilde{f}(t_i) = \sum_{n=0}^{N-1} c_n h_n(t_i T) \qquad (15)$$

The time scaling factor in this case is T = 0.586 and the approximate signal energy, as calculated by the coefficients of the DOGHT transform c_n and the Parseval relation (11), is : $\tilde{E} = 1.97069$ while the exact signal energy as calculated by relation (12) is : $E = 2$.

From Table 1 it is evident that the inverse DOGHT transform is exact and that in this case its odd coefficients are zero because of the even symmetry of the hyperbolic secant function. In Table 2 the respective results for the DCT transform are shown. It is seen that the DCT transform approximates less accurately the signal energy.

However, the most important fact is that the DOGHT achieves much better concentration of signal energy in the lower order coefficients (lower “frequencies”) and thus it can lead to better compression. For example, in the case of the zero order coefficient, we have $c_0 = 0.39$ for the DCT (DC coefficient) and $c_0 = 1.06$

for the DOGHT (coefficient of resemblance with the Gaussian function).

Table 1. DOGHT – secht. $\tilde{E}=1.97069$

i	$\frac{t_i}{T}$	$f\left(\frac{t_i}{T}\right)$	c_i	$\tilde{f}\left(\frac{t_i}{T}\right)$
0	-5.00	0.01	1.06	0.01
1	-3.38	0.07	0.00	0.07
2	-1.97	0.27	-0.18	0.27
3	-0.65	0.82	0.00	0.82
4	0.65	0.82	0.10	0.82
5	1.97	0.27	0.00	0.27
6	3.38	0.07	-0.03	0.07
7	5.00	0.01	0.00	0.01

Table 2. DCT – secht. $\tilde{E}=1.94486$

i	$\frac{t_i}{T}$	$f\left(\frac{t_i}{T}\right)$	c_i	$\tilde{f}\left(\frac{t_i}{T}\right)$
0	-5.00	0.01	0.39	0.01
1	-3.57	0.06	0.00	0.06
2	-2.14	0.23	-0.39	0.23
3	-0.71	0.79	0.00	0.79
4	0.71	0.79	0.18	0.79
5	2.14	0.23	0.00	0.23
6	3.57	0.06	-0.07	0.06
7	5.00	0.01	0.00	0.01

An important disadvantage of the DOGHT transform is that the sampling points required for its computation are not uniformly distributed, while in all practical applications the sampled signals are acquired through uniform sampling. In order to overcome this obstacle, it is proposed here, to use interpolation and more specifically linear or quadratic interpolation. In such a way, starting from the signal values at the uniformly sampled points we obtain through interpolation the signal values at the non-uniformly sampled points that are required for the computation of the DOGHT transform coefficients. This interpolation procedure introduces an additional error that is however negligible in most practical cases and is compensated rapidly by the faster convergence of the DOGHT.

The interpolation used is in the general case a Lagrange type interpolation, and more specifically for n=2 points (linear interpolation):

$$f(t)=L_0(t)f(t_0)+L_1(t)f(t_1),$$
$$L_0(t)=\frac{t-t_1}{t_0-t_1}, \qquad L_1(t)=\frac{t-t_0}{t_1-t_0} \qquad (16)$$

and for n = 3 points (quadratic interpolation) :

$$f(t)=L_0(t)f(t_0)+L_1(t)f(t_1)+L_2(t)f(t_2),$$
$$L_0(t)=\frac{(t-t_1)(t-t_2)}{(t_0-t_1)(t_0-t_2)}, L_1(t)=\frac{(t-t_0)(t-t_2)}{(t_1-t_0)(t_1-t_2)},$$
$$L_2(t)=\frac{(t-t_0)(t-t_1)}{(t_2-t_0)(t_2-t_1)} \qquad (17)$$

Thus, the detailed steps followed by the compression method of this paper are :

(a) Input N signal level data from uniform sampling $f(t_k)$ in the time interval $[-t_{\max}, t_{\max}]$

(b) Computation of the time scaling factor: $T=\frac{t_{N-1}}{t_{\max}}$

(c) Computation of the signal level $f\left(\frac{t_i}{T}\right)$ at the N points: $\frac{t_i}{T}$ (where t_i are the zeroes of the N-th order Hermite polynomial) through linear (16) or quadratic interpolation (17)

(d) Computation of the N coefficients c_n of the DOGHT transform by using relation (1)

(e) Retention of the N1<N coefficients c_n in order to achieve N/N1 times compression.

Example 2

As an example we will compare the compression of the signal $f(t)=\exp(-|t|)$ in the time window $t\in[-5,+5]$ with N = 16 points from the DCT and the DOGHT with quadratic interpolation. In order to compare the two methods we will use the normalised rms error calculated at the points of the uniform sampling (PRD – Percent RMS Difference) :

$$PRD=100*\left[\frac{\sum_{k=0}^{N-1}\left[f(t_k)-\tilde{f}(t_k)\right]^2}{\sum_{k=0}^{N-1}\left[f(t_k)\right]^2}\right]^{1/2}$$

where t_k are the points of the uniform sampling, and the maximum absolute error (PE – Peak amplitude Error):

$$PE = \max\{error\},$$

$$error = \left| f(t_k) - \tilde{f}(t_k) \right|$$

Table 3 represents the results of the compression of the two methods for compression ratios ranging from 1 :1 to 4 :1. It is easily seen that the DOGHT transform achieves 4 :1 compression with less rms error (approximately 13 %) than the DCT transform rms error for only 2 :1 compression (approximately 14 %).

Table 3. Percent RMS error (PRD)

N1	DCT	DOGHT
16	0.0 %	1.22 %
12	4.40 %	1.47 %
8	14.29 %	3.80 %
4	43.37 %	12.91 %

Tables 4 and 5 present an extensive comparison of the two transforms. FF(tk) is the inverse transform calculated from the limited coefficients $c_n, n = 0, K, N1-1$, i.e. the reconstruction of the signal F(t) after decompression.

It is also seen here that the maximum absolute error PE is somewhat lower for the DOGHT, although at a double compression ratio than the DCT, PE = 0.07, while the maximum absolute error for the DCT is: PE = 0.08. Also PRD = 12.91% for the DOGHT and PRD = 14.29% for the DCT, showing the superiority of the DOGHT at higher compression ratios.

Table 4. 16-point DCT method with 2:1 compression (Tmax = 5, N1 = 8)

K	tk	f(tk)	cn	ff(tk)	error
1	-5.000	0.007	0.259	-0.016	0.02
2	-4.333	0.013	0.000	0.030	0.02
3	-3.667	0.026	-0.273	0.057	0.03
4	-3.000	0.050	0.000	0.039	0.01
5	-2.333	0.097	0.149	0.052	0.04
6	-1.667	0.189	0.000	0.196	0.01
7	-1.000	0.368	-0.084	0.451	0.08
8	-0.333	0.717	0.000	0.658	0.06
9	0.333	0.717	0.049	0.658	0.06
10	1.000	0.368	0.000	0.451	0.08
11	1.667	0.189	-0.029	0.196	0.01
12	2.333	0.097	0.000	0.052	0.04
13	3.000	0.050	0.017	0.039	0.01
14	3.667	0.026	0.000	0.057	0.03
15	4.333	0.013	-0.008	0.030	0.02
16	5.000	0.007	0.000	-0.016	0.02

Table 5. 16-point DOGHT method with 4:1 compression (Tmax = 5, N1 = 4, T = 0.938)

k	tk	f(tk)	cn	ff(tk)	error
1	-5.000	0.007	0.918	0.000	0.01
2	-4.333	0.013	0.000	0.000	0.01
3	-3.667	0.026	-0.033	0.001	0.02
4	-3.000	0.050	0.000	0.008	0.04
5	-2.333	0.097	0.112	0.049	0.05
6	-1.667	0.189	0.000	0.183	0.01
7	-1.000	0.368	-0.019	0.436	0.07
8	-0.333	0.717	0.000	0.670	0.05
9	0.333	0.717	0.036	0.670	0.05
10	1.000	0.368	0.000	0.436	0.07
11	1.667	0.189	-0.009	0.183	0.01
12	2.333	0.097	0.000	0.049	0.05
13	3.000	0.050	0.013	0.008	0.04
14	3.667	0.026	0.000	0.001	0.02
15	4.333	0.013	-0.002	0.000	0.01
16	5.000	0.007	0.000	0.000	0.01

Example 3

As another example we will compare the compression of the pseudo-random signal $f(t) = \exp(-|t|) + 0.1*(random - 0.5)$ in the time window $t \in [-5,+5]$ with N = 16 points from the DCT and the DOGHT with quadratic interpolation, where random are pseudo-random numbers in the interval $t \in [0,1]$. Table 6 represents the results of the compression of the two methods for compression ratios ranging from 1:1 to 4:1. It is easily seen, also in this case, that the DOGHT transform achieves 4:1 compression with less rms error (approximately 17%) than the DCT transform rms error for only 2:1 compression (approximately 18 %). Tables 7 and 8 present an extensive comparison of the two transforms. It is seen again in this case that the maximum absolute error PE is lower for the DOGHT, PE = 0.09, while PE = 0.10 for the DCT.

Table 6. Percent RMS error (PRD)

N1	DCT	DOGHT
16	0.0 %	3.57 %
12	7.73 %	7.54 %
8	17.72 %	10.54 %
4	44.51 %	16.75 %

Table 7. 16-point DCT method with 2:1 compression (Tmax = 5, N1=8)

k	tk	f(tk)	cn	ff(tk)	error
1	-5.000	0.038	0.260	-0.006	0.04
2	-4.333	-0.021	-0.005	0.030	0.05
3	-3.667	0.016	-0.259	0.047	0.03
4	-3.000	0.080	-0.011	0.030	0.05
5	-2.333	0.057	0.150	0.056	0.00
6	-1.667	0.232	0.007	0.210	0.02
7	-1.000	0.358	-0.073	0.460	0.10
8	-0.333	0.719	-0.004	0.646	0.07
9	0.333	0.685	0.052	0.630	0.06
10	1.000	0.340	0.014	0.425	0.09
11	1.667	0.168	-0.029	0.186	0.02
12	2.333	0.128	0.023	0.053	0.07
13	3.000	0.027	0.022	0.040	0.01
14	3.667	0.023	0.001	0.061	0.04
15	4.333	0.061	-0.015	0.049	0.01
16	5.000	0.027	-0.017	0.020	0.01

Table 8. 16-point DOGHT method with 4:1 compression (Tmax = 5, N1 = 4, T = 0.938)

k	tk	f(tk)	cn	ff(tk)	error
1	-5.000	0.038	0.897	0.000	0.04
2	-4.333	-0.021	0.025	0.000	0.02
3	-3.667	0.016	-0.025	0.000	0.02
4	-3.000	0.080	-0.013	0.007	0.07
5	-2.333	0.057	0.112	0.049	0.01
6	-1.667	0.232	-0.001	0.191	0.04
7	-1.000	0.358	-0.026	0.448	0.09
8	-0.333	0.719	0.003	0.664	0.05
9	0.333	0.685	0.049	0.639	0.05
10	1.000	0.340	-0.045	0.407	0.07
11	1.667	0.168	0.000	0.176	0.01
12	2.333	0.128	-0.016	0.053	0.07
13	3.000	0.027	0.046	0.011	0.02
14	3.667	0.023	0.043	0.002	0.02
15	4.333	0.061	0.011	0.000	0.06
16	5.000	0.027	-0.007	0.000	0.03

3. Conclusion

Consequently, the conclusion drawn from the above simple examples is that the compression properties of the DOGHT transform are superior from these of the DCT transform and this for a variety of signals. Thus, it can be predicted that the use of the DOGHT transform will lead to an important increase of the compression ratios of biomedical images and this at a computational complexity that is similar to the computational complexity of the DCT transform. Future scope of this work is to investigate in detail, the performance of the proposed compression method for dermatology images.

References

[1] N. Ahmed, T. Natarajan, K.R. Rao , Discrete Cosine Transform, IEEE Transactions on Computers, Vol.C-23, No.1, 1974, pp. 90-93.

[2] D.J. Bailey, N. Birch, Image compression using a DCT image processor, Electronic Engineering, Vol.61, No.751, 1989, pp. 41-44.

[3] M. Antonini, M. Barlaud, P. Mathieu, I. Daubechies, Image coding using the wavelet transform, IEEE Transactions on Image Processing, Vol.1, 1992, pp. 205-220.

[4] C.A. Christopoulos, T. Ebrahimi, A.N. Skodras, JPEG2000: The new still picture compression standard, ACM Multimedia Workshop, Marina del Rey CA, USA, 2000, pp. 45-49.

[5] P. Lazaridis, G. Debarge, P. Gallion, Discrete Orthogonal Gauss-Hermite transform for optical pulse propagation analysis, Journal Optical Society America B, Vol.20, No.7, 2003, pp.1508-1513.

[6] P. Lazaridis, G. Debarge, P. Gallion, Split-Step-Gauss-Hermite algorithm for fast and accurate simulation of soliton propagation, International Journal Numerical Modeling, Vol.14, 2001, pp. 325-329.

[7] P. Lazaridis, G. Debarge, P. Gallion, Exact solutions for linear propagation of chirped pulses using a chirped Gauss-Hermite orthogonal basis, Optics Letters, Vol.22, No.10, 1997, pp. 685-687.

[8] A.J. Jerri, The application of general discrete transforms to computing orthogonal series and solving boundary value problems, Bulletin Calcutta Mathematical Society, Vol.71, 1979, pp. 177-187.

This work is supported by the project ARCHIMEDES II, EPEAEKII: sub-project 2.2.3 and it is co-funded by the EU and the Greek Government.