is about 8600. In our simulation, the design runs at 66-MHz clock frequency, and achieves a search rate of about 683 000 blocks/s.

III. Experimental Results and Performance Comparisons

In our experiments, the block size is fixed at 16×16 and the maximum motion displacement is set to ± 7. The first 100 frames of the "Football," "Mobile," "Windmill," "Flower," "Tennis," "Salesman," and "Miss America" sequences are used to test the proposed algorithm. It is noted that these sequences contain different combinations of still, slow, and fast moving objects. The comparisons are made by using ten search algorithms: 1) FS; 2) 3SS; 3) CS; 4) PHODS; 5) 4SS; 6) BBGDS; 7) SES; 8) PSA; 9) GPS; and 10) FPS, in terms of six different measures. These measures include: 1) average MSE per pixel; 2) average PSNR; 3) average prediction errors per pixel; 4) average entropy of prediction errors; 5) average percentage of unpredictable pels per frame (pels with absolute prediction errors larger than three, over a range of 255, are classified as unpredictable pels [3]); and 6) average search points per block. The results obtained from the different algorithms are reported in Tables II–VII. These results indicate that the proposed FPS performs better than other algorithms in terms of the six measures for the seven image sequences. Although FPS performs just a little better than GPS, its cost is less because it needs only a small memory (128 bytes) for fuzzy prediction, as compared with the large memory (32 000 bytes) needed for GPS [8].

Fig. 7 shows the original 15th frame of the "Football" sequence, and the motion-compensated prediction frames using FS, 3SS, 4SS, BBGDS and FPS, respectively. The compensated images look unsatisfactory at the locations where fast moving objects appear. From Fig. 7, we find that only FS and FPS can clearly identify the number "82" of the football player who slipped on the ground. In terms of subjective image quality, the performance of FPS is better than those of 3SS, 4SS, and BBGDS. Table VIII shows the results of the six measures of FS, 3SS, 4SS, BBGDS, and FPS for the 15th frame of the Football sequence.

IV. Conclusion

In this paper, we present a novel and efficient fuzzy-prediction search algorithm for block-motion estimation. Experimental results show that the proposed algorithm works better than other search algorithms. The VLSI architecture for the proposed algorithm has been developed, and yields a search rate of 683 000 blocks/s with a clock rate of 66 MHz in our simulation.

References

[1] T. Koga, K. Iinuma, A. Iijima, and T. Ishiguro, "Motion-compensated interframe coding for video conferencing," in *Proc. NTC81*, New Orleans, LA, 1981, pp. C9.6.1–9.6.5.

[2] M. Ghanbari, "The cross-search algorithm for motion estimation," *IEEE Trans. Commun.*, vol. 38, pp. 950–953, July 1990.

[3] L.-G. Chen, W.-T. Chen, Y.-S. Jehng, and T.-D. Chiueh, "An efficient parallel motion estimation algorithm for digital image processing," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 1, pp. 378–385, Apr. 1991.

[4] L.-M. Po and W.-C. Ma, "A novel four-step search algorithm for fast block motion estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, pp. 313–317, Mar. 1996.

[5] L.-K. Liu and E. Feig, "A block-based gradient descent search algorithm for block motion estimation in video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, pp. 419–422, Apr. 1996.

[6] J. Lu and M. L. Liou, "A simple and efficient search algorithm for block-matching motion estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, pp. 429–433, Feb. 1997.

[7] L. Luo, C. Zou, X. Gao, and Z. He, "A new prediction search algorithm for block motion estimation in video coding," *IEEE Trans. Consumer Electron.*, vol. 43, pp. 56–61, 1997.

[8] J. M. Jou, P.-Y. Chen, and J-M. Sun, " The grey prediction search algorithm for block motion estimation," *IEEE Trans. Circuits Syst. Video Technol.*, to be published

[9] L. A. Zadeh, "Fuzzy sets," *Inform. Control*, vol. 8, no. 6, pp. 338–353, 1965.

[10] C. C. Lee, "Fuzzy logic in control systems: Fuzzy logic controller-Part I & Part II," *IEEE Trans. Syst., Man, Cybern.*, vol. 20, pp. 404–435, Feb. 1990.

A New Discrete Fractional Fourier Transform Based on Constrained Eigendecomposition of DFT Matrix by Largrange Multiplier Method

Soo-Chang Pei, Chien-Cheng Tseng, and Min-Hung Yeh

***Abstract*—This paper is concerned with the definition of the discrete fractional Fourier transform (DFRFT). First, an eigendecomposition of the discrete Fourier transform (DFT) matrix is derived by sampling the Hermite Gauss functions, which are eigenfunctions of the continuous Fourier transform and by performing a novel error-removal procedure. Then, the result of the eigendecomposition of the DFT matrix is used to define a new DFRFT. Finally, several numerical examples are illustrated to demonstrate that the proposed DFRFT is a better approximation to the continuous fractional Fourier transform than the conventional defined DFRFT.**

***Index Terms*—Discrete Fourier transform, discrete fractional Fourier transform.**

I. Introduction

In recent years, a new fractional operator called fractional Fourier transform (FRFT) has been investigated extensively [1]–[11]. The FRFT has found many applications in the solution of differential equations, quantum mechanics and quantum optics, and optical systems and optical signal processing, swept-frequency filters, time-variant filtering and multiplexing, pattern recognition, and study of time-frequency distribution [1]–[4]. Besides, the FRFT has been proven to relate to other signal analysis tools, such as Wigner distribution, neural network, wavelet transform and various chirp-related operations [5]–[7]. Several useful properties of FRFT are currently under study in the signal processing community [8]–[11].

So far, many methods for implementing FRFT have been developed. However, most of them utilize optical instruments or numerical integration. Because the FRFT is a potentially useful tool for signal

Manuscript received April 20, 1998; revised June 16, 1999. This paper was recommended by Associate Editor P. S. A. Diniz.

S.-C. Pei is with the Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan, R.O.C.

C.-C. Tseng is with the Department of Computer and Communication Engineering, National Kaohsiung First University of Technology and Science, Kaohsiung, Taiwan, R.O.C.

M.-H. Yeh is with the Department of Computer Information Science, Tamui Oxford University College, Tamsui, Taipei, Taiwan, R.O.C.

Publisher Item Identifier S 1057-7130(99)08035-0.

TABLE I
MULTIPLICITIES OF $\mathbf{F}$

N	Mul. of 1	Mul of -1	Mul. of -j	Mul. of j
4m	m+1	m	m	m-1
4m+1	m+1	m	m	m
4m+2	m+1	m+1	m	m
4m+3	m+1	m+1	m+1	m

processing, the direct computation of FRFT in digital computers has become an important issue. Basically, the computation of the discrete fractional Fourier transform (DFRFT) needs to obey the additivity property and similarity condition. The additivity property means that application of the transform with angular parameter α followed by an application of the transform with angular parameter β is equivalent to the application of the transform with angular parameter $\alpha+\beta$. The similarity condition means that the transform results of DFRFT must be similar to those of the continuous FRFT. In [12] and [13], a method for digital computing FRFT was proposed, but their method does not obey the additivity property. In [14], another DFRFT is defined, but this definition does not provide the similar transform results as those of continuous case. The purpose of this paper is to present a new DFRFT which obeys the additivity property and similarity condition simultaneously.

The advantage to define a DFRFT which satisfies additivity property is described as follows. If additivity property holds, the inverse transform of DFRFT with angular parameter α is the DFRFT with angular parameter $-\alpha$. Thus, the signals processed in the fractional Fourier domains can be easily transformed into time domain by a unified computer program with minus angular parameter. The DFRFT proposed in [12], [13] does not obey the additivity property, so its inverse transform has not been described yet. Moreover, the reasons to define a DFRFT which obeys the similarity condition are that the continuous signal processing algorithms derived in continuous fractional Fourier domains can be directly modified into the digital signal cases by replacing continuous FRFT with DFRFT. In the sequel, the chirp interference filtering applications will be used to illustrate this fact [7].

II. EIGENDECOMPOSITION OF THE DISCRETE FOURIER TRANSFORM (DFT) MATRIX

A. The Eigenvalues of DFT Matrix

Now, we review the properties of the eigenvalues of the DFT matrix $\mathbf{F}$ whose elements are defined by

$$\mathbf{F}_{nk} = \frac{1}{\sqrt{N}}\left(\cos\left(\frac{2\pi kn}{N}\right) - j\,\sin\left(\frac{2\pi kn}{N}\right)\right), \qquad 0 \le n,\, k \le N-1. \tag{1}$$

From the results in [15], the properties of eigenvalues of DFT matrix can be summarized by the following fact.

Fact 1: The eigenvalues of $\mathbf{F}$ are $\{1, -1, j, -j\}$ and its multiplicities are listed in Table I.

(Pf) see [15]: From this fact, it is clear that there exists infinite eigendecomposition forms of the DFT matrix, because any linear combination of the eigenvetors which correspond to the same eigenvalue is also an eigenvector. If we use the eigendecomposition of the DFT matrix $\mathbf{F}$ to define DFRFT, then we have infinite choice. However, under the condition that transform results of DFRFT needs to be similar to those of continuous FRFT, the eigendecomposition of DFT matrix must be found trickily. In the following, we will derive an eigendecomposition form by sampling the Hermite Gauss functions, which are the eigenfunctions of the continuous Fourier transform and by performing a novel error removal procedure. Using the proposed decomposition to define DFRFT, the transform results will obey similar conditions.

B. An Eigendecomposition of DFT Matrix

The usual continuous Fourier transform pair is defined as

$$\mathbf{X}(\omega) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} x(t)e^{-j\omega t}\,\mathrm{dt}$$
$$x(t) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} \mathbf{X}(\omega)e^{j\omega t}\,d\omega.$$

It can be shown that the eigenfunctions of the Fourier transformation operator are Hermite Gauss function $H_m(t)e^{-t^2/2}$, where $H_m(t)$ are the Hermite polynomials of order m defined by

$$H_m(t) = \sum_{k=0}^{\lfloor m/2\rfloor} \frac{(-1)^k m!(2t)^{m-2k}}{k!}. \tag{2}$$

The notation $\lfloor m/2\rfloor$ denotes the largest integer smaller than $m/2$. We thus have

$$(-j)^m H_m(\omega)e^{-\omega^2/2} = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} H_m(t)e^{-t^2/2}e^{-j\omega t}\,\mathrm{dt}. \tag{3}$$

Now, we will use this equation to derive an approximate eigendecomposition of the DFT matrix. Our derivation is mainly based on the following two facts.

Fact 2: If the sequence $g_m(n)$ is obtained by sampling the Hermite Gauss function $H_m(t)e^{-t^2/2}$ with sampling interval $T = \sqrt{2\pi/N}$, i.e.,

$$g_m(n) = H_m(nT)e^{-(nT)^2/2} \tag{4}$$

then it can be shown that

$$(-j)^m g_m(k) \approx \sqrt{\frac{1}{N}}\sum_{n=-N/2}^{N/2-1} g_m(n)e^{-j(2\pi kn/N)} \tag{5}$$

for sufficiently large N.

Proof: By truncating the integral interval of the (3) from $(-\infty, \infty)$ to $(-NT/2, NT/2)$, we have the following approximation expression:

$$(-j)^m H_m(\omega)e^{-\omega^2/2} \approx \frac{1}{\sqrt{2\pi}}\int_{-NT/2}^{NT/2} H_m(t)e^{-t^2/2}e^{-j\omega t}\,\mathrm{dt}. \tag{6}$$

This approximation is valid because $NT = \sqrt{2\pi N}$ is big for large N and the decay rate of Gauss function $e^{(-t^2)/2}$ is very fast. Next, by replacing the continuous integral with numerical integral, we have

$$\int_{-NT/2}^{NT/2} H_m(t)e^{-t^2/2}e^{-j\omega t}\,\mathrm{dt} \approx \sum_{n=-N/2}^{N/2-1} H_m(nT)e^{-(nT)^2/2}e^{-j\omega nT}T. \tag{7}$$

This approximation is also valid bacause $T = \sqrt{2\pi/N}$ is very small when N is very large. Combining (6) and (7), we obtain

$$(-j)^m H_m(\omega)e^{-\omega^2/2} \approx \sqrt{\frac{1}{N}}\sum_{n=-N/2}^{N/2-1} H_m(nT)e^{-(nT)^2/2}e^{-j\omega n\sqrt{2\pi/N}}. \tag{8}$$

This expression is valid for any ω. Thus, take $\omega = kT = k\sqrt{2\pi/N}$ at both sides, we have

$$(-j)^m H_m(kT)e^{-(kT)^2/2} \approx \sqrt{\frac{1}{N}} \sum_{n=-N/2}^{N/2-1} H_m(nT)e^{-(nT)^2/2}e^{-j2\pi kn/N}. \quad (9)$$

Substitute (4) into (9), and the proof is completed.

From the above proof, it is clear that there are two approximation errors in (5). One is truncation error in (6), the other is numerical error in (7). When the value of N approaches infinity, both errors approach to zero. Thus, the larger N value is, the better approximation (5) is. Next, because the degree of Hermite polynomial $H_m(t)$ is m, the decay rate of the Hermite Gauss function $H_m(t)e^{-t^2/2}$ is proportional to $t^m e^{-t^2}$ for sufficiently large t. Thus, the larger order m is, the slower decay rate Hermite Gauss function has. This implies that the truncation error in (6) is large for large order m. Thus, when order m becomes large, the approximation in (5) becomes inaccurate.

Fact 3: If the sequence $\overline{g}_m(n)$ defined in the range $[0, N-1]$ is obtained by shifting Hermite Gauss samples $g_m(n)$ defined in the range $[-N/2, N/2-1]$ in the following way:

$$\overline{g}_m(n) = \begin{cases} g_m(n), & \text{for } 0 \le n \le \dfrac{N}{2} - 1 \\ g_m(n-N), & \text{for } \dfrac{N}{2} \le n \le N-1 \end{cases} \quad (10)$$

then it can be shown that the DFT of the $\overline{g}_m(n)$ can be approximated by $(-j)^m\overline{g}_m(k)$, i.e.,

$$(-j)^m\overline{g}_m(k) \approx \sqrt{\frac{1}{N}} \sum_{n=0}^{N-1} \overline{g}_m(n)e^{-j(2\pi kn/N)} \quad (11)$$

for sufficiently large N.

Proof: Using the equality $e^{-j(2\pi nk/N)} = e^{-j[2\pi(n-N)k/N]}$, this fact can be proven trivially.

From Fact 3, it is clear that $\overline{g}_m(n)$ are the approximate eigenfunctions of the discrete Fourier transform. Because the Hermite Gauss functions are orthogonal to each other for different orders, the sequences $\overline{g}_{m1}(n)$ and $\overline{g}_{m2}(n)$ are approximately orthogonal for $m1 \neq m2$, i.e.,

$$\sum_{n=0}^{N-1} \overline{g}_{m1}(n)\overline{g}_{m2}(n) \approx 0. \quad (12)$$

Let us define the vectors $\mathbf{v}_m$ as

$$\mathbf{v}_m = [\overline{g}_m(0)\overline{g}_m(1) \cdots \overline{g}_m(N-1)]^t. \quad (13)$$

Then, (11) means that

$$(-j)^m\overline{\mathbf{v}}_m \approx \mathbf{F}\overline{\mathbf{v}}_m \quad (14)$$

where $\overline{\mathbf{v}}_m = \mathbf{v}_m/\|\mathbf{v}_m\|$ is normalized version of the vector $\mathbf{v}_m$. Thus, $\overline{\mathbf{v}}_m$ is an approximate eigenvector of the DFT matrix $\mathbf{F}$ corresponding to the eigenvalue $(-j)^m$. Although the approximate expression in (14) is valid for any order m, the DFT matrix $\mathbf{F}$ with size $N \times N$ has only N eigenvectors whose eigenvalues need to satisfy the multiplicity property in Fact 1. Thus, we are required to select N orders denoted by the set $\Psi = \{m_1, m_2, \cdots, m_N\}$ $(m_1 < m_2 < \cdots < m_N)$ to construct an eigendecomposition of the matrix $\mathbf{F}$. Two rules of the selection in this paper are listed as follows.

1) The set $\{(-j)^{m_1}, (-j)^{m_2}, \cdots, (-j)^{m_N}\}$, formed by eigenvalues, must satisfy the multiplicity property in Fact 1.
2) The approximation error $\|(-j)^{m_i}\overline{\mathbf{v}}_{m_i} - \mathbf{F}\overline{\mathbf{v}}_{m_i}\|$ must be less than the error $\|(-j)^m\overline{\mathbf{v}}_m - \mathbf{F}\overline{\mathbf{v}}_m\|$ if m is not in the set Ψ.

TABLE II
A SUITABLE CHOICE OF SET Ψ OBEYING TWO RULES

N	$\Psi = \{m_1, \cdots, m_N\}$
4n	$0, 1, 2, \cdots, 4n-2, 4n$
4n+1	$0, 1, 2, \cdots, 4n-1, 4n$
4n+2	$0, 1, 2, \cdots, 4n, 4n+2$
4n+3	$0, 1, 2, \cdots, 4n+1, 4n+2$

Because the approximation error $\|(-j)^m\overline{\mathbf{v}}_m - \mathbf{F}\overline{\mathbf{v}}_m\|$ becomes large when order m increases, a suitable choice of set Ψ which obeys two rules is described in Table II.

Based on this choice, an approximation eigendecomposition of the DFT matrix $\mathbf{F}$ is given by

$$\mathbf{F} \approx \sum_{i=1}^{N} (-j)^{m_i}\overline{\mathbf{v}}_{m_i}\overline{\mathbf{v}}_{m_i}^t. \quad (15)$$

In order to remove the error in this decomposition, an eigenvector calibration procedure is developed as follows. Assume that the eigenvector set $\{\overline{\mathbf{v}}_{m_1}, \overline{\mathbf{v}}_{m_2}, \cdots, \overline{\mathbf{v}}_{m_N}\}$ will be corrected into the eigenvector set $\{\mathbf{u}_{m_1}, \mathbf{u}_{m_2}, \cdots, \mathbf{u}_{m_N}\}$ and the vectors from $\mathbf{u}_{m_1}$ to $\mathbf{u}_{m_{k-1}}$ have been obtained. Then, the eigenvector $\mathbf{u}_{m_k}$ is found by minimizing the squared error $(\mathbf{u}_{m_k} - \overline{\mathbf{v}}_{m_k})^2$ subjected to two prescribed constraints which are the eigenvector constraint $\mathbf{F}\mathbf{u}_{m_k} = (-j)^{m_k}\mathbf{u}_{m_k}$ and the orthogonal constraint $\mathbf{u}_i^t\mathbf{u}_{m_k} = 0$ for $i = m_1, \cdots, m_{k-1}$. After some manipulation, the two constraints can be written in matrix form as

$$\mathbf{C}_{m_{k-1}}\mathbf{u}_{m_k} = \mathbf{0} \quad (16)$$

where the matrix $\mathbf{C}_{m_{k-1}}$ is given by

$$\mathbf{C}_{m_{k-1}} = \begin{bmatrix} \mathrm{Real}(\mathbf{F} - (-j)^{m_k}\mathbf{I}) \\ \mathrm{Img}(\mathbf{F} - (-j)^{m_k}\mathbf{I}) \\ \mathbf{u}_{m_1}^t \\ \vdots \\ \mathbf{u}_{m_{k-1}}^t \end{bmatrix}. \quad (17)$$

The notation $\mathrm{Real}(.)$ and $\mathrm{Img}(.)$ denote the real part and imaginary part of a complex matrix, and $\mathbf{I}$ is identity matrix. Using the QR decomposition, the matrix $\mathbf{C}_{m_{k-1}}$ can be rewritten as

$$\mathbf{C}_{m_{k-1}} = \mathbf{Q}_{m_{k-1}}\begin{bmatrix} \mathbf{R}_{m_{k-1}} \\ \mathbf{0} \end{bmatrix}. \quad (18)$$

Substitute (18) into (16), (16) reduces to

$$\mathbf{R}_{m_{k-1}}\mathbf{u}_{m_k} = \mathbf{0}. \quad (19)$$

If the rank of matrix $\mathbf{C}_{m_{k-1}}$ is equal to r, the size of the matrix $\mathbf{R}_{m_{k-1}}$ is $r \times N$. Now, using the well-known Largrange multiplier method, the solution of this constrained optimization problem is given by

$$\mathbf{u}_{m_k} = (\mathbf{I} - \mathbf{R}_{m_{k-1}}^t(\mathbf{R}_{m_{k-1}}\mathbf{R}_{m_{k-1}}^t)^{-1}\mathbf{R}_{m_{k-1}})\overline{\mathbf{v}}_{m_k}. \quad (20)$$

Finally, the entire eigenvector calibration procedure is summarized as follows: Given DFT matrix $\mathbf{F}$ and the approximate eigenvector set $\{\overline{\mathbf{v}}_{m_1}, \overline{\mathbf{v}}_{m_2}, \cdots, \overline{\mathbf{v}}_{m_N}\}$ we take the following steps to compute the exact eigenvector set $\{\mathbf{u}_{m_1}, \mathbf{u}_{m_2}, \cdots, \mathbf{u}_{m_N}\}$.

Step 1: Let matrix $\mathbf{C}_{m_1}$ be $[\mathbf{I} - \mathrm{Real}(\mathbf{F})^t, \mathrm{Img}(\mathbf{F})^t]^t$ and use (20) to find the vector $\mathbf{u}_{m_1}$. Note that we normalize $\mathbf{u}_{m_1}$ to unit norm. Set $k = 2$.

Step 2: Perform the following two computations:

a) Use (17), (18) to compute the matrix $\mathbf{R}_{m_k}$.
b) Use (20) to calculate the vector $\mathbf{u}_{m_k}$ and normalize it to unit norm.

Step 3: Let $k = k + 1$. If $k > N$, stop the procedure. Otherwise go to Step 2.

After this calibration, the exact eigendecomposition of the DFT matrix $\mathbf{F}$ is given by

$$\mathbf{F} = \sum_{i=1}^{N} (-j)^{m_i} \mathbf{u}_{m_i} \mathbf{u}_{m_i}^t. \tag{21}$$

The unique feature of this eigendecomposition is that the shape of the eigenvector is similar to the shape of the Hermite Gauss functions which is the eigenfunction of the continuous Fourier transform. In the next section, we will use this decomposition to define a discrete fractional Fourier transform.

III. New Definition of DFRFT

A. Continuous FRFT

The continuous FRFT is defined as [8]

$$F^{\alpha}[x(t)] = \int_{-\infty}^{\infty} x(t) K_{\alpha}(t, \omega)\, \mathrm{d}t \tag{22}$$

where the transform kernel is given by (23), shown at the bottom of the page. After some manipulation, it is easy to show that

$$F^{\alpha+\beta}[x(t)] = F^{\alpha}[x(t)] F^{\beta}[x(t)]. \tag{24}$$

This implies that the angle additivity property is satisfied, i.e., application of the transform with angular parameter α followed by an application of the transform with angular parameter β is equivalent to the application of the transform with angular parameter $\alpha + \beta$. Moreover, the Hermite Gauss functions are also the eigenfunctions of the continuous fractional Fourier transform, i.e.,

$$\begin{aligned} F^{\alpha}[H_m(t) e^{-t^2/2}] &= (-j)^{2m\alpha/\pi} H_m(\omega) e^{-\omega^2/2} \\ &= e^{-j\alpha m} H_m(\omega) e^{-\omega^2/2}. \end{aligned} \tag{25}$$

It is clear that the eigenvalue of the mth order Hermite Gauss function is $e^{-j\alpha m}$. When $\alpha = \pi/2$, the continuous FRFT becomes the conventional Fourier transform. As an example, Fig. 1(a) shows the FRFT of the impulse function $\delta(t)$ for the angle $\alpha = 0.45\pi$. The continuous FRFT of this special signal has the closed form formula given by

$$F^{\alpha}[\delta(t)] = \sqrt{\frac{1 - j\cot(\alpha)}{2\pi}}\; e^{j(\omega^2/2)\cot(\alpha)}. \tag{26}$$

The real parts of FRFT or DFRFT in this paper are plotted by solid lines, and the imaginary parts of FRFT or DFRFT are plotted by dashed or dotted lines.

B. Conventional Discrete Fractional Fourier Transform

Let data vector be $\mathbf{x}$; Santhanam and McClellan defined the discrete fractional Fourier transform as [14]

$$\Re^{\alpha}[\mathbf{x}] = \mathbf{F}^{2\alpha/\pi} \mathbf{x}. \tag{27}$$

The $2\alpha/\pi$th power of the DFT matrix $\mathbf{F}$ is found by the equation

$$\mathbf{F}^{2\alpha/\pi} = \sum_{k=0}^{3} a_k(\alpha) \mathbf{F}^k \tag{28}$$

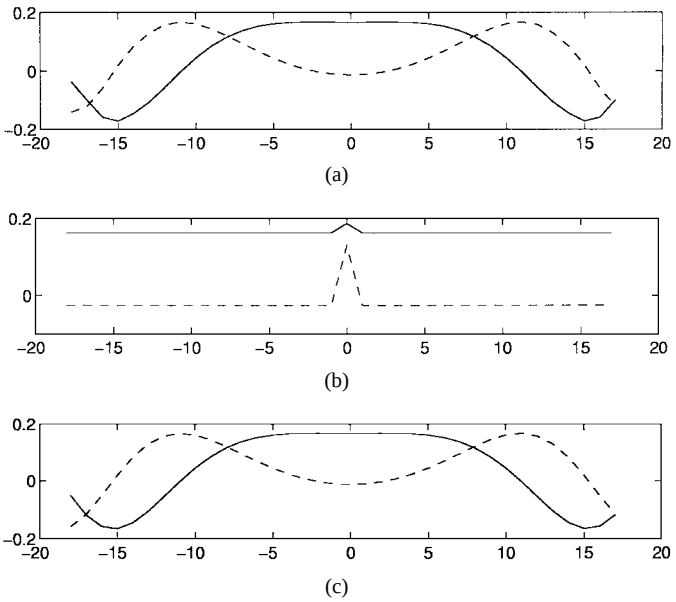

Fig. 1. (a) Continuous FRFT of the impulse signal $\delta(t)$. (b) Conventional DFRFT of the impulse function $\delta(n)$. (c) Proposed DFRFT of the impulse function $\delta(n)$ for $\alpha = 0.45\pi$.

where the coefficients $a_i(\alpha)$ are given in [14]. Although this definition of DFRFT obeys the angle additivity property, it is not the discrete version of the continuous transform defined in (23). A numerical example is illustrated as follows. Fig. 1(b) shows the result of the DFRFT produced by a discrete impulse function defined as

$$\delta(n) = \begin{cases} 1, & \text{for } n = 0 \\ 0, & \text{otherwise.} \end{cases} \tag{29}$$

In this example, we choose $N = 36$. The results shown in Fig. 1(b) are far from the results in Fig. 1(a).

C. New Discrete Fractional Fourier Transform

As (27), the DFRFT of the data vector $\mathbf{x}$ is defined by

$$\Re^{\alpha}[\mathbf{x}] = \mathbf{F}^{2\alpha/\pi} \mathbf{x}.$$

Since $2\alpha/\pi$th power of the DFT matrix $\mathbf{F}$ can be calculated from its eigendecomposition by taking the $2\alpha/\pi$th power for its eigenvalues, the matrix $\mathbf{F}^{2\alpha/\pi}$ is given by

$$\mathbf{F}^{2\alpha/\pi} = \sum_{i=1}^{N} (-j)^{m_i(2\alpha/\pi)} \mathbf{u}_{m_i} \mathbf{u}_{m_i}^t. \tag{30}$$

Because $(-j)^{m_i(2\alpha/\pi)} = e^{-jm_i\alpha}$, the eigenvalues of the new transform matrix $\mathbf{F}^{2\alpha/\pi}$ are consistent with those of the continuous FRFT. Moreover, the eigenvectors $\mathbf{u}_{m_i}$ are obtained by sampling Hermite Gauss functions with an error removal procedure, so the eigenvectors of new DFRFT are similar to those of the continuous FRFT. Due to these two agreements, the transform result of our DFRFT will be similar to that of continuous FRFT. Fig. 1(c) shows the result of the new DFRFT produced by a discrete impulse function defined in (29). It is clear that the shape of the transformed result

$$K_{\alpha}(t, \omega) = \begin{cases} \sqrt{\dfrac{1 - j\cot\alpha}{2\pi}}\; e^{j((t^2+\omega^2)/2)\cot\alpha - j\omega t \csc\alpha}, & \text{if } \alpha \text{ is not a multiple of } \pi \\ \delta(t - \omega), & \text{if } \alpha \text{ is a multiple of } 2\pi \\ \delta(t + \omega), & \text{if } \alpha + \pi \text{ is a multiple of } 2\pi \end{cases} \tag{23}$$

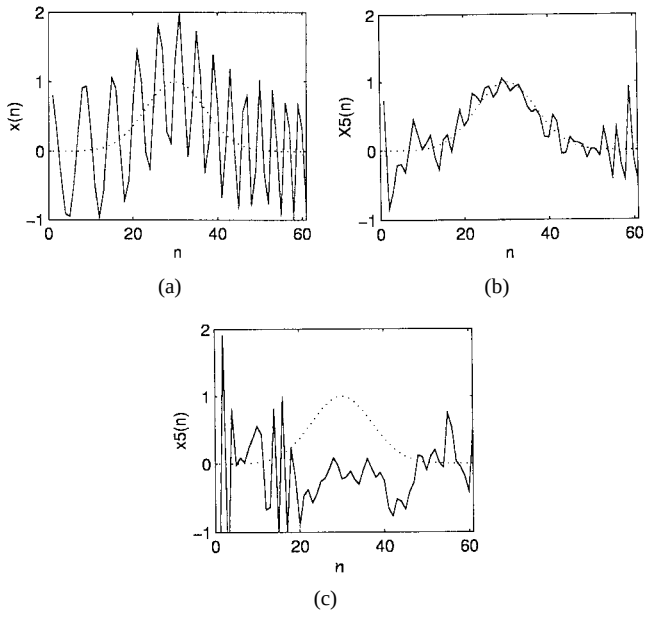

Fig. 2. (a) Waveform of signal $x(n)$ which is Gaussian signal with chirp interference. (b) Waveform of the signal x_5 filtered by the proposed DFRFT. (c) Waveform of the signal x_5 filtered by the DFRFT in [14]. The waveform plotted by dotted line is the ideal Gaussian signal.

looks much like the shape of the continuous transformed result in Fig. 1(a).

In order to evaluate the similarity between the results of continuous FRFT in Fig. 1(a) and the results of DFRFT in Fig. 1(b) and (c), a quantitative error measure is defined as follows:

$$E = \sqrt{\sum_{n=-N/2}^{N/2-1} \|F_C^{\alpha}(nT) - F_D^{\alpha}(n)\|^2} \tag{31}$$

where $F_C^{\alpha}(nT)$ is the sequence obtained by directly sampling continuous FRFT with the sampling interval $T = \sqrt{2\pi/N}$, and $F_D^{\alpha}(n)$ is the transformed result of DFRFT. It is clear that the smaller error E is, the more similar results between continuous FRFT and DFRFT have. Now, the error E between Fig. 1(a) and (b) is 1.2815, and the error E between Fig. 1(a) and (c) is 0.0407. Thus, the proposed DFRFT provides a better similarity than the conventional DFRFT defined in [14].

D. Chirp Filtering in the DFRFT Domain

In the following, we concentrate on the applications of the chirp interference removal. The detail of continuous chirp case has been investigated in [7]. Here, we only extend the technique developed in [7] to the discrete chirp case. Since the FRFT of chirp signal is the line delta function in the appropriate fractional Fourier domain, we can remove this impulse of chirp component in FRFT domain by multiplying a narrow band-stop mask. The narrower the band-stop mask is, the less distortion the nonchirp part has. Given the angular parameter $\tau = 2\alpha/\pi$ and the signal $x(n)$ composed of a desired signal and a chirp interference, the procedure of filtering out this chirp component in DFRFT domain is summarized as follows.

Step 1: Compute the DFRFT $x_1(n)$ of the signal $x(n)$ with angular parameter τ.

Step 2: Multiply the transform result $x_1(n)$ by the band-stop mask $m(n)$. The masking result is denoted by $x_2(n) = x_1(n)m(n)$.

Step 3: Compute the DFRFT $x_3(n)$ of the signal $x_2(n)$ with angular parameter -2τ.

Step 4: Multiply the result $x_3(n)$ by the band-stop mask $m(n)$. The masking result is denoted by $x_4(n) = x_3(n)m(n)$.

Step 5: Compute the DFRFT $x_5(n)$ of the signal $x_4(n)$ with angular parameter τ. The signal $x_5(n)$ is the desired filtered output.

The above procedure is the same as that of the continuous case except signal is discrete and continuous FRFT is replaced with DFRFT. Two band-stop masks are applied in the procedure because a real chirp signal can be decomposed into the sum of two complex chirp signals. One band-stop mask is performed in the τ domain, the other is in the $-\tau$ domain. In the following, we use the above procedure to filter a Gaussian signal with a real chirp noise, that is

$$x(n) = e^{-0.01(n-30)^2} + \cos(0.004\pi n^2 + 0.2\pi n), \quad 1 \le n \le 61. \tag{32}$$

Fig. 2(a) shows the waveform of signal $x(n)$ plotted by solid line. For comparison, the ideal Gaussian signal $e^{-0.01(n-30)^2}$ is also plotted by dotted line in Fig. 2(a)–(c). The angular parameter τ is chosen as 0.9 and the band-stop mask is given by

$$m(n) = \begin{cases} 0, & \text{for } 40 \le n \le 48 \\ 1, & \text{otherwise.} \end{cases} \tag{33}$$

Fig. 2(b) and (c) shows the filtered signals $x_5(n)$ with solid line by using the proposed DFRFT and Santhanam's DFRFT in [14], respectively. From the results in Fig. 2(b), it is clear that the chirp interference has been eliminated by the proposed DFRFT. The Gaussian signal to chirp noise ratio is improved from -3.8 to 6.5 dB. However, the results in Fig. 2(c) show that the chirp interference can not be removed by the DFRFT in [14]. This is owing that the proposed DFRFT obeys similarity condition. Thus, the continuous signal-processing algorithms derived in continuous fractional Fourier domains can be directly modified into the discrete signal cases by replacing continuous FRFT with the proposed DFRFT, which satisfies the similarity condition.

IV. Conclusion

In this paper, a new definition of the DFRFT based on an eigendecomposition of the DFT matrix has been presented. However, the complexity for implementing DFRFT is $O(N^2)$, which is the same as that of DFT. Thus, it is interesting to develop a fast algorithm to compute DFRFT. This topic will be investigated in the future.

References

[1] V. Namias, "The fractional order Fourier transform and its application to quantum mechanics," *J. Inst. Math. Applicat.*, vol. 25, pp. 241–265, 1980.

[2] H. M. Ozaktas and B. Barshan, "Convolution, filtering, and multiplexing in fractional Fourier domains and their relationship to chirp and wavelet transform," *J. Opt. Soc. Amer. A*, vol. 11, pp. 547–559, 1994.

[3] A. W. Lohmann, "Image rotation, Wigner rotation, and the fractional Fourier domain," *J. Opt. Soc. Amer. A*, vol. 10, pp. 2181–2186, 1993.

[4] D. Mendlovic, Z. Zalevsky, R. G. Dorsch, Y. Bitran, A. W. Lohmann, and H. Ozaktas, "New signal representation based on the fractional Fourier transform: Definitions," *J. Opt. Soc. Amer. A*, vol. 11, pp. 2424–2431, 1995.

[5] D. Dragoman, "Fractional Wigner distribution function," *J. Opt. Soc. Amer. A*, vol. 13, pp. 474–478, 1996.

[6] S. Y. Lee and H. H. Szu, "Fractional Fourier transforms, wavelet transform, and adaptive neural network," *Opt. Eng.*, vol. 33, pp. 2326–2330, 1994.

[7] R. G. Dorsch, A. W. Lohmann, Y. Bitran, and D. Mendlovic, "Chirp filtering in the fractional Fourier domain," *Appl. Opt.*, vol. 33, pp. 7599–7602, 1994.

[8] L. B. Almeida, "The fractional Fourier transform and time frequency representation," *IEEE Trans. Signal Processing*, vol. 42, pp. 3084–3091, Nov. 1994.

[9] H. M. Ozaktas and O. Aytar, "Fractional Fourier domains," *Signal Processing*, vol. 46, pp. 119–124, 1995.
[10] H. M. Ozaktas, N. Erkaya, and A. Kutay, "Effect of fractional Fourier transform on time-frequency distributions belonging to the Cohen class," *IEEE Signal Processing Lett.*, vol. 3, pp. 40–41, Feb. 1996.
[11] X. G. Xia, "On bandlimited signals with fractional Fourier transform," *IEEE Signal Processing Lett.*,, vol. 3, pp. 72–74, Mar. 1996.
[12] M. A. Kutay, H. M. Ozaktas, L. Onural, and O. Arikan, "Optimal filtering in fractional Fourier domains," in *Proc. ICASSP*, 1995, pp. 937–940.
[13] H. M. Ozaktas, O. Arikan, A. Kutay, and G. Bozdagi, "Digital computation of the fractional Fourier transform," *IEEE Trans. Signal Processing*, vol. 44, pp. 2141–2150, Sept. 1996.
[14] B. Santhanam and J. H. McClellan, "The DRFT—A rotation in time frequency space," in *Proc. ICASSP*, 1995, pp. 921–924.
[15] J. H. McClellan and T. W. Parks, "Eigenvalue and eigenvector decomposition of the discrete Fourier transform," *IEEE Trans. Audio Electroacoust.*, vol. AE-20, pp. 66–74, Mar. 1972.

An Adiabatic Differential Logic for Low-Power Digital Systems

Chun-Keung Lo and Philip C. H. Chan

***Abstract*—A new adiabatic circuit technique called adiabatic differential cascode voltage switch with complementary pass-transistor logic tree (ADCPL) is presented. ADCPL is a dual-rail logic with relatively low gate complexity. It operates from a two-phase nonoverlapping supply clock. Power reduction is achieved by recovering the energy in the recover phase of the supply clock. Energy dissipation comparison with other logic circuits is performed. Simulation shows that for a pipelined ADCPL carry lookahead adder, a power reduction of 50%–70% can be achieved over the static complimentary metal oxide semiconductor case within a practical operation frequency range. The results also show that the lower the operating frequency, the larger the energy savings for an ADCPL circuit.**

***Index Terms*—Adiabatic circuit, differential cascode voltage logic, low-power circuit, low-power digital system.**

I. Introduction

Adiabatic switching is a new approach for reducing power dissipation in digital logic. When adiabatic switching is used, the signal energies stored on circuit capacitances may be recycled instead of dissipated as heat [1]. For an energy recovery circuit, the ideal energy dissipation when a capacitance C is charged from zero to $V_{\rm dd}$ or discharged from $V_{\rm dd}$, through a circuit of resistance R during time T is given by

$$E_{\rm diss} = I^2RT = \left(\frac{CV_{\rm dd}}{T}\right)^2 RT = \left(\frac{RC}{T}\right)CV_{\rm dd}^2. \qquad (1)$$

When $T \gg RC$, the energy dissipation is much smaller than the conventional complimentary metal oxide semiconductor (CMOS) circuit, for which an energy of $(CV_{\rm dd}^2)/2$ is required during a charge or discharge cycle. As a result, when the charging time T is much larger than the RC time constant, the power consumption can be reduced. Usually, this can be achieved by using a slowly changing clocking waveform to drive the circuit.

Manuscript received September 2, 1998; revised July 1, 1999. This work was supported by the Research Grants Council under Grant HKUST 547/94E. This paper was recommended by Associate Editor N. Ranganathan.

The authors are with the Department of Electrical and Electronic Engineering, The Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong.

Publisher Item Identifier S 1057-7130(99)08031-3.

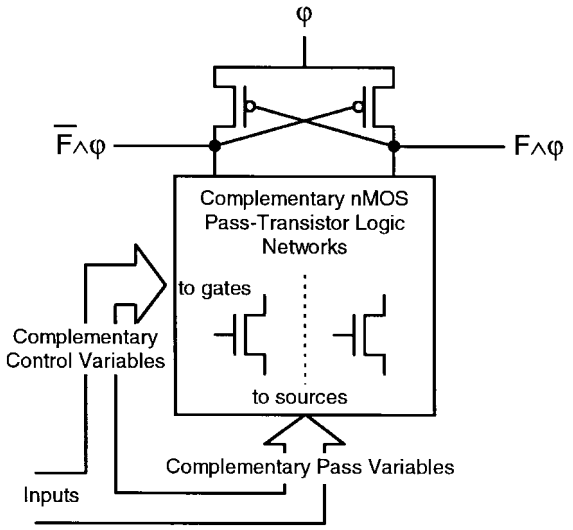

Fig. 1. Basic ADCPL circuit configuration.

A variety of adiabatic logic architectures has been proposed for low power VLSI design [2]–[5]. Most of them use diodes or diode-like devices for precharge, which causes unavoidable energy loss due to the voltage drop across the diodes. Other designs have been proposed to eliminate the precharge diode; however, they have potential problems of floating output nodes and faulty logic [5]. Previous studies [1]–[5] have shown that the most important operational characteristics for the successful utilization of adiabatic logic architectures are a constant load presented by the gates to the clock, the reduction or elimination of floating output nodes, the reduction of the number of clock phases needed for correct operation, the ability to generate a signal and its complement on the same clock phase, the reduction or elimination of diodes, the ability to drive a high output to $V_{\rm dd}$ and a low output to $V_{\rm ss}$, and finally an efficient supply clock generator to provide the power clock.

This paper describes a new adiabatic circuit technique called adiabatic differential cascode voltage switch with complementary pass-transistor logic tree (ADCPL). With complementary pass-transistor logic (CPL) as the logic evaluation tree, the input capacitances can be reduced and complex logic function can be realized efficiently [6]. ADCPL is diode free and is a dual-rail logic operated with a two-phase nonoverlapping supply clock. Energy is recovered in the recover phase of the supply clock, and pipeline structure can be designed in conjunction with the clock-powered latch. The basic ADCPL logic design and operation will be presented. Energy dissipation comparison with other adiabatic logic and static CMOS logic is made. A ripple carry adder (RCA) and a pipelined carry lookahead adder (CLA) using ADCPL are constructed to evaluate the performance.

II. Logic Design and Operation

The basic ADCPL circuit configuration is shown in Fig. 1. It has a similar structure to the differential cascode voltage switch (DCVS) logic [7], but with a complementary pass-transistor logic (CPL) tree instead of the nMOS logic tree in the conventional DCVS circuit.