博 士 学 位 论 文

# 信号处理中的快速离散变换算法研究


博士生姓名 吴一全

学科、专业 通信与电子系统

研究方向 信息获取与处理

指导教师 朱兆达教授

南京航空航天大学



一九九八年四月

博 士 学 位 论 文

# 信号处理中的快速离散变换算法研究


博士生姓名 吴一全

学科、专业 通信与电子系统

研 究 方 向 信息获取与处理

指 导 教 师 朱兆达 教授



南 京 航 空 航 天 大 学

一九九八年六月

# RESEARCH ON FAST ALGORITHMS FOR THE DISCRETE TRANSFORM IN SIGNAL PROCESSING

**A**

**DISSERTATION**

Submitted to Graduate School of
Nanjing University of Aeronautics and Astronautics

for the Degree of

**DOCTOR OF PHILOSOPHY**

by

**Yiquan Wu**

Supervisor: Prof. ***Zhaoda Zhu***

June 1998

# 摘　　要

快速离散变换在信号处理中占有举足轻重的地位。它是数字信号处理中起着支柱作用的基本工具，对数字信号处理学科发展和广泛应用起着重要作用。其应用范围也和数字信号处理一样广泛，遍及科学和工程的许多领域，如雷达、图像处理与识别、声纳、声学、语音识别、仪器仪表、生物医学工程、地震信号处理、振动分析与机械故障诊断等。因此，研究快速离散变换具有重要的理论意义与广泛的实用价值。

算法研究是快速离散变换研究最主要的方面。一般按算法研究的国际通行标准，以实乘次数、实加次数及实乘和实加总次数作为衡量算法性能的主要指标。

第一章阐明了信号处理中一系列快速离散变换的提出过程，展示了这些快速离散变换在信号处理其它方面的应用。依据大量的有关文献，分门别类较全面地综述了离散 Fourier 变换(DFT)与实离散 Fourier 变换(RDFT)，离散余弦变换(DCT)与离散正弦变换(DST)、离散 Hartley 变换(DHT)与离散 W 变换(DWT)这三类离散变换快速算法的发展概况。

第二章研究有关离散 Fourier 变换与实离散 Fourier 变换的快速算法。提出以下五种算法：(1)一种新的实乘子 FFT-j 算法；(2)ODFT 的复共轭乘子分离基快速算法；(3)基于双线性内插曲面逼近的二维 Fourier 变换的一种高精度快速算法；(4)基于 FFT 的一维与二维信号快速内插算法；(5)利用实数 LMS 自适应滤波器计算 RDFT 与 DFT。

第三章研究有关离散 Hartley 变换与离散 W 变换的快速算法。提出以下五种算法：(1)长度 $2^m$ 快速 Hartley 变换的一种算法；(2)分离基快速 W 变换(-Ⅱ，Ⅲ)算法；(3)基 3FFT 与 FHT 的新算法；(4)基 3/9 快速 Hartley 变换的新算法；(5)基于 FHT 的离散信号快速内插算法。

第四章研究有关离散余弦变换与离散正弦变换的快速算法。提出以下五种算法：(1)分离基快速余弦变换算法；(2)利用快速 W 变换-j 计算离散余弦变换-Ⅳ的新算法；(3)离散正弦变换-Ⅱ的快速递归算法；(4)二维矢量基离散正弦变换的快速算法；(5)基于快速 W 变换-Ⅱ计算离散正弦变换-Ⅳ的新算法。

第五章研究二维图像矩的快速算法。提出两种新算法。一种是计算二维图像矩的神经网络方法；另一种是计算图像矩的高精度快速算法。

第六章结束语。列出本文研究工作的主要贡献，指出可进一步研究的若干问题。

本文的内容几乎均已在有关学术刊物上发表或待发表。

**关键词**：数字信号处理，离散变换，快速算法，离散 Fourier 变换，实离散 Fourier 变换，离散 Hartley 变换，离散 W 变换，离散余弦变换，离散正弦变换，图像处理，图像矩

**中图分类号**：TN911.7；TN27；O174.22

# ABSTRACT

Fast discrete transforms play a decisive role in signal processing. They are the most fundamental operations in digital signal processing. They have played a key role in the development and widespread use of digital signal processing in a variety of applications like radar, image processing and recognition, sonar, acoustics, speech recognition, scientific instrumentation, biomedical electronics, seismic processing. Hence, to research into fast discrete transforms is of great theoretical significance and of practical value.

Fast algorithms are pricipal for the discrete transforms. Generally the number of multiplication operations and addition operations required for fast discrete transforms is considered as the major quality index.

This thesis is organized as follows.

Chapter 1 presents the history of ideas on fast discrete transforms and their applications in other fields of signal processing. The development of the fast algorithms is surveyed for three types of discrete transforms (discrete Fourier transform (DFT) and real discrete Fourier transform (RDFT), discrete cosine transform (DCT) and discrete sine transform (DST), and discrete Hartley transform (DHT) and discrete W transform) according to a large number of literatures. The content of this thesis are outlined.

Chapter 2 considers the fast algorithms for the discrete Fourier transform and real discrete Fourier transform concerned. Five algorithms are proposed as follows. (1) A new real-multiplier FFT-j algorithm; (2) A split-radix fast algorithm for the ODFT with complex conjugate multiplier; (3) A high precision fast algorithm for the two-dimensional Fourier transform; (4) Fast interpolation algorithms for signals in one and two dimensions; (5) Computation of the RDFT and DFT through the real LMS adaptive filter.

Chapter 3 treats of five fast algorithms for the discrete Hartley transform and discrete W transform. (1) A new fast Hartley transform algorithm; (2) The split-radix fast W transform(—Ⅱ. Ⅲ)algorithms; (3) New algorithms for radix-3 FFT and FHT; (4) Radix 3/9 fast Hartley transform; (5) A fast interpolation algorithm for discrete signals based on FHT.

In chapter 4, five fast algorithms for the discrete cosine transform and discrete sine transform are posed. (1) The split-radix fast cosine transform algorithm; (2) A new algorithm for computing the discrete cosine transform-Ⅳ using fast W transform-j; (3) A fast recursive algorithm for the discrete sine transform-Ⅱ; (4) A two-dimensional vector-radix fast algorithm for the discrete sine transform; (5) A new fast algorithm for computing the discrete sine transform-Ⅳ by fast W transform-Ⅱ.

Chapter 5 deals with two fast algorithms for the two-dimensional image moments.

One is a neural net approach to computation of two-dimensional image moments, the other is a high precision fast algorithm for the two-dimensional image moments.

Finally, chapter 6 outlines the main contribution of this thesis. Some problems to be solved in future studies are indicated.

**Keywords**: digital signal processing, discrete transform, fast algorithm, discrete Fourier transform, real discrete Fourier transforms, discrete Hartley transform, discrete W transform, discrete cosine transform, discrete sine transform, image processing, image moments.

# 目　　录

# 第一章 绪 论

## 1.1 快速离散变换的提出及其在信号处理中的应用

### 1.1.1 快速离散变换的提出

变换在信号处理中占有举足轻重的地位。但在信号处理中只有那些具有快速算法的离散变换才能得到实际应用。线性滤波和 Fourier 变换是最基本的运算,然而,直接计算将给信号处理带来沉重的计算负担。1965 年 Cooley 和 Tukey 发表了关于计算离散 Fourier 变换(DFT)的标志性论文[1],所提出的算法使计算 DFT 的时间减少了几个数量级。这篇论文的发表拉开了信号处理中快速离散变换算法研究的序幕,引起了信号处理界的强烈反响,随之掀起研究热潮。根据 DFT 的卷积定理,这一算法也可直接用来快速计算线性卷积与相关。因此,这种后来被称为快速 Fourier 变换(FFT)的算法,对数字信号处理学科发展和广泛应用所起的作用,如何形容都不为过。为纪念 FFT 的提出,普及 FFT 技术,1990 年 Signal Processing 杂志特别刊登了 Duhamel 和 Vetterli 撰写的一篇长达 40 页的 FFT 述评[64]。IEEE Signal Processing Magazine 也于 1992 年和 1994 年先后发表了特约文章[2],并出版专辑[3]。

几乎在提出 FFT 的同时,1966 年 Taki 和 Hatori 把 Hadamard 于 1893 年研究而 Walsh 于 1923 年提出的 Walsh-Hadamard 变换(WHT)首次用于通信系统获得成功。因 WHT 不需乘法,易于硬件实现,促使人们去开拓应用领域。1969 年 Pratt 等人应用 WHT 进行图像编码的再次成功,激励人们尝试或寻求各种快速离散正交变换。七十年代初,研究以 WHT 为代表的非正弦类变换风气浓厚。Haar 变换、修改的 WHT、广义变换、斜变换等一系列非正弦类变换相继提出[4],并用于图像编码。直至发现了离散余弦变换,人们才对 WHT 及其它非正弦类变换的研究热情逐渐衰退。这是因为非正弦类变换无论在性能或便于理论分析方面均远不如正弦类变换,且随着计算机技术和 VLSI 工艺的发展,非正弦类变换易于实现的优势也逐步丧失。

1974 年,Ahmed 等人首次发现了离散余弦变换(即 DCT-Ⅱ)[5]。两年后 Hamidi 和 Pearl 证明了 DCT-Ⅱ的去相关性优于 DFT[6]。1976 年 Jain 提出了离散正弦变换-Ⅰ(DST-Ⅰ)[7]。1978 年 Kekre 和 Solanki 首次提到 DST-Ⅱ和 DST-Ⅲ[8],并用于图像压缩。1979 年 Jain 从统计最佳变换 Karhunen-Loeve 变换(KLT)的逆矩阵出发,研究了一组正弦类变换(其中有 DCT-IV 和 DST-IV)[9],认为对于一阶平稳马尔可夫过程,DST-Ⅱ、DST-Ⅰ和 DCT-Ⅱ分别在相关系数 $\rho$ 的取值范围 $-1<\rho<-0.5$,$|\rho|<0.5$ 和 $0.5<\rho<1$ 三种条件下最逼近 KLT。事实上,$\rho\to-1,0$ 和 1 时 KLT 分别退化为 DST-Ⅱ,DST-Ⅰ和 DCT-Ⅱ[10,11]。1980 年 Kitajima 提出了对称 DCT 即 DCT-Ⅰ[12],认为在适当加窗后,它比 DCT-Ⅱ更接近 KLT。随后的检验说明,$\rho$ 取中等值或 $N$ 较大时,DCT-Ⅰ最接近 KLT[13,14]。1992 年 Ersoy 等从信号的对称性延拓出发,导出了一种新的非正交 DCT,称之为 DC3T[15]。它相当于在时域先对信号预处理再作 $N+1$ 点 DCT-Ⅰ。1995 年吴乐南证明了它具有与 DCT-Ⅱ相同的残余相关和更高的变换编码增益[16]。一般来说,DCT 和 DST 的提出是为了寻求既有快速算法

又逼近 KLT 的变换，目的是去相关、滤噪声、压缩次要分量或抽取主要特征。其直接应用领域为语音与图像编码、数据压缩、变换域自适应滤波与神经计算及模式识别中的特征抽取与选择等。已有的各种成熟的图像压缩编码标准如 JPEG、MPEG-1、MPEG-2、H. 261 以及 HDTV 等都无一例外地采用 DCT/IDCT[201]。

为了使实序列经 DFT 后，仍能变为实数，1983 年 Ersoy 提出了实离散 Fourier 变换(RDFT)[17]。它对应于三角函数形式的 Fourier 级数，如同 DFT 对应于复指数形式的 Fourier 级数。RDFT 是一种实正交变换，用 RDFT 计算实序列循环卷积比用 DFT 更为有效。

离散 Hartley 变换(DHT)或离散 W 变换(DWT)则是为了在实数域内取代 DFT 的作用而提出的。早在 1942 年 Hartley 就提出了一种正反变换核相同的实变换，但随后近四十年里，仅有两本书用三言两语提及 Hartley 的工作。1981 年王中德重新提出这一变换，称之为 W 变换[18-20]。1983 年底 Bracewell 给出这一变换的离散形式，称之为离散 Hartley 变换(DHT)[21]。1984 年 Bracewell 将 Cooley-Tukey 算法移植到 DHT 上[22]。与此同时，王中德定义了四种类型离散 W 变换(DWT)[23]，随后论证了 DWT 的性质[24]。其中 DWT-Ⅰ 即为 DHT。DHT 或 DWT 的提出，立即引起人们的关注并受到日益重视。主要原因是 DHT(DWT)是一种仅涉及实数运算的实正交变换，DHT(DWT-Ⅰ)的正反变换核形式完全相同，而且 DHT 或 DWT 与 DFT、DCT 及 DST 的关系极为密切，快速 Hartley 变换(FHT)可完全取代实值 FFT 快速计算循环卷积与相关。1994 年 Proc. IEEE 出版了关于 Hartley 变换的专辑[25]，可见这种变换已在实际应用中得到广泛重视。王中德在定义四种 DWT 时，同时把 DCT、DST 也各划分成四种类型[23,24]。考虑到 DCT、DST、DWT 各自均有四种类型，1991 年我们也相应把具有一定对称特性与反对称特性的广义离散 Fourier 变换(GFT)[26] 划分成四种类型，即通常标准的 DFT 定为 DFT-Ⅰ，把奇 DFT(ODFT)[27]定为 DFT-Ⅱ 和 DFT-Ⅲ，而把奇平方 DFT($O^2$DFT)[28]划成 DFT-Ⅳ。这不仅统一了这些不同寻常的 DFT，而且所划分的四类 DFT 与 DCT、DST、DWT 的四种类型完全对应起来(1994 年 Ersoy 也作了和我们同样的划分)[29]。由此一个比较完整的正弦类快速离散变换家族建立起来。

### 1.1.2 快速离散变换在信号处理中的应用

快速离散变换算法在数字信号处理中占据重要地位。由于离散变换是数字信号处理中起着支柱作用的基本工具，而采用快速算法可以减少计算机处理时间，提高计算精度，节省存贮单元数目，简化数字硬件的设计与开发，降低芯片的成本和尺寸，使芯片性能得到显著改善。因此，快速离散变换算法应用范围也和数字信号处理的应用范围一样广泛，遍及科学和工程的许多领域，如雷达、图像处理、声纳、声学、语音处理、通信、多媒体计算机、仪器仪表、生物医学工程、模式识别、地震信号处理、遥感、振动分析与机械故障诊断、电磁学等。这些应用领域对快速算法的需要不断增长。若仅从快速离散变换算法在信号处理本身这一学科中的应用来看，可归纳如表 1.1。

表 1.1 快速离散变换在信号处理中的应用

| | |
|---|---|
| 图像处理 | 图像变换编码、图像变换域增强、<br>图像恢复、纹理分析、边界与区域描述 |
| 信号复原与重建 | 带限信号内插与外推、解卷积、<br>计算成像(SAR 成像、CT 成像等) |
| 自适应滤波,<br>非线性滤波 | 变换域自适应滤波、同态滤波、复倒谱、微分倒谱、<br>非线性自适应滤波 |
| 谱估计,<br>信号检测与辨识 | 经典与现代谱估计、频谱细化、求解 Toeplitz 方程组、<br>匹配滤波、时延估计、时频分析、高阶谱 |
| 模式识别<br>神经网络 | 特征抽取与选择,<br>变换域神经计算 |

## 1.2 快速离散变换算法的发展概况

算法研究是快速离散变换发展最主要的方面。由于 DFT 和 RDFT、DCT 和 DST、DHT 或 DWT 的快速算法分别在不同时期沿着不同的途径发展,我们分门别类加以概述。

### 1.2.1 DFT 和 RDFT 快速算法的发展概况

**1. DFT 快速算法的发展**

DFT 快速算法的发展大致可以分为三个时期、四个热点。

第一时期(六十年代中期~七十年代中期):序列长度 $N=2^m$ 的 Cooley-Tukey 类算法是当时的热点。自从 1965 年 Cooley 和 Tukey 的论文发表后,为提高算法的可用性,紧接着有几篇论文发表。强调方法的双分性,导致基 2 按频率抽取算法[30];以提高基数达到减少运算量的目的,Bergland 提出了高基算法(如基 4、基 8 甚至基 16 算法)[31],但基数提高,运算量减少幅度迅速下降,算法复杂性急剧增加。Bergland 还给出了实序列基 8 算法子程序[32]。与此同时,Singleton 提出了适合任意长度的混合基算法[33]。Rabiner 等人提出了线性调频(Chirp)Z 变换算法[211]。Bluestein 提出了计算 DFT 的 Chirp 滤波法[212]。

第二时期(七十年代中期~八十年代中期):这一时期有两个热点,其一是序列长度 N 分解成几个互素因子的 Good 类算法[34],其二是序列长度 $N=2^m$ 点实乘子算法。其中 Good 类算法占据主导地位。这时,可应用 Good 映射,把一维 DFT 映射成多维 DFT[35,36]。而 Rader 曾证明过如何把一个 $N$ 点 DFT 映射成长度为 $N-1$ 的循环卷积[37]。1976 年 Winograd 首次把 Good 和 Rader 的结果有效地组合起来[38,39]。将小 $N$ 点 DFT 转化成循环卷积,并用高效算法计算循环卷积,通过把小 $N$ 点 DFT 嵌套成大 $N$ 点 DFT,从而使乘法次数大大减少[40]。该算法被称为 Winograd Fourier 变换算法(WFTA)。如果不用嵌套,而采用行列法计算多维 DFT,所得算法称之为素因子算法(PFA)[41,42]。与 WFTA 相比,1977 年提出的 PFA 所需乘法次数较多,但加法次数少,且结构较 WFTA 简单,但二者均比基 2 算法复杂,WFTA 尤为复杂。要真正弄懂 WFTA 和 PFA 算法,需具备较多数学知识。WFTA 的复杂性理

论结果完美，一旦实现达不到预期效果。Morris 对 WFTA 和基 4FFT 在通用机上作了一个比较，结论却是基 4FFT 比 WFTA 效率高[43]。

在这一时期，长度 $N=2^m$ 的算法研究仍在持续。1976 年 Rader 和 Brenner 或许受 WFTA 的启发，提出了一种实乘子算法[44]。该算法通过用实乘子或纯虚乘子取代复乘子使乘法次数显著减小，其代价是增加了加法次数和对噪声的敏感性。因此，Preuss 和 Cho、Temes 又对该算法作了进一步研究，提出了另外两种实乘子算法[45,46]，但均没达到减少加法次数的目的。1993 年我们提出的实(或纯虚)乘子快速算法成功地达到了这一目的，该算法是目前运算量最少的算法之一且结构非常简单。1990 年黄顺吉给出了按频率抽取的 Rader-Brenner FFT 算法矩阵分解公式[188]，但主要公式有误。笔者已予修正，并给出了运算量更少的分解公式。此外，1978 年 Bruun 提出了仅用实运算计算实序列 DFT 的多项式分解法[213]。1979 年 Nakayama 提出了一种时、频域混合抽取 FFT 算法[189]。1982 年被推广到二维[190]。

第三时期(八十年代中期～九十年代)：研究热点又转回 $N=2^m$ 类算法并研究 $N=3^m$ 的情况。1984 年 Duhamel 和 Hollmann 提出了分离基 FFT(SRFFT)算法[47]。所得算法结构相对简单，与以往长度 $N=2^m$ 算法相比，该算法所需乘法次数和加法次数均为最少。同年，Martens 提出的割圆分解法(RCFA)[48]，Vetterli 和 Nussbaumer 提出的以 DCT-Ⅱ 为媒介的 FFCT 算法[49]从不同角度出发也达到了与 SRFFT 算法同样的运算量。随后两年，王中德和 Suehiro 等人分别对 1984 年提出的对称分解法[23]作了改进也基本达到这一水平[50,51]。1991 年 Gupta 和 Rao 仿照 FFCT 算法给出了 FFST 算法，所需运算量与 FFCT 算法相同[191]。另一方面，Duhamel 继提出复序列 SRFFT 后，又给出了实序列与实对称序列的 SRFFT 算法[52]。Sorensen 紧接着总结性地论述了有效计算实序列 DFT 的各种快速算法[53]，包括基 2 实 FFT、基 4 实 FFT、实 SRFFT、实 PFA 和实 WFTA。1989 年 Vetterli 和 Duhamel 将长度 $2^m$ 分离基算法推广到长度 $p^m$[54]，证明了基 $p/p^2$ 算法将优于基 $p$ 和基 $p^2$ 算法，并研究了长度为 $3^m$DFT 的基 3/9 算法。1993 年我们提出了基 3DIF 与 DIT FFT 新算法，其实乘数比 Dubois-Venetsanopoulos 基 3 算法[55]少 $3N+1$。我们随后提出的基 3/9FFT 算法更优于 Vetterli 和 Duhamel 提出的基 3/9 算法[54]和 1994 年 Stasiniski 提出的基 3/9 算法[57]。

近几年因为许多先进的工作站和信号处理芯片设计成有效的融合乘法和加法的运算，基本运算为 $\pm a \pm bc$，称为乘/加运算($m/a\text{-}op$)。为充分利用这一特点，1993 年 Liner 和 Feig 提出了融合乘/加运算的 FFT 分度算法[56]。把复旋转因子变为平凡乘法运算。此外，1994 年 Stasiniski 提出了针对其它基的基 K-FFT 算法[57]。它由 $K$ 点卷积和 $K$ 点 DFT 构成。文中列举了基 3、基 6、基 2/4、基 2/6、基 3/6 算法的结构和计算复杂性。

针对实际中仅需少数 DFT 输出点或仅有几个输入值不为零这两种特殊情况，1993 年 Sorensen 提出用短 FFT 算法和 Goertzel 算法组合的方法[58]。该方法优于通常的 FFT 修枝法[59-61,192-194]。我们已将该算法推广到 DFT-Ⅱ，Ⅲ，Ⅳ 和二维情况。

值得一提的是，1987 年 Widrow 等人提出了利用复数 LMS 自适应算法计算 DFT 的方法[62]。该方法为 DFT 的并行计算与 VLSI 实现提供了与前述 FFT 不同的另一手段。1993 年 Liu 和 Bruton 已将该算法推广到二维[63]。我们提出了利用实数 LMS 自适应算法计算复序列 DFT 的方法，所需运算量只有 Widrow 算法的一半。

由于多维 DFT 需大量运算和存贮空间，人们对此也进行了仔细研究。一般分成四类算

法[64]：行列法、矢量基(基 2，基 4)算法[65,66,195-197]、分离矢量基算法[67,68,198]和多项式变换算法。其中令人感兴趣的算法是 1978 年由 Nussbaumer 与 Quandalle 首先提出的多项式变换解法[69,70]。这类算法所需运算量最少，一维奇 DFT(ODFT)的计算是其中的关键之一。我们提出的 ODFT 复共轭乘子分离基快速算法是目前所需运算量最少的 ODFT 算法，比 Pei 和 Luo1996 年提出的算法早四年多，且优于 Pei 和 Luo 的算法[200]。需要指出的是，1986 年 Heideman 和 Burrus 证明了计算长度 $2^m$ DFT 所需的理论上最小实乘次数是 $2^{m+2}-2m^2-2m-4$[199]，目前所有计算长度 $2^m$DFT 的算法所需实乘次数均未达到该理论最小值。

2. RDFT **快速算法的发展**

至于 RDFT 快速算法的发展则十分简单。1988 年 Ersoy 和 Hu 仿照 FFT 给出了计算 RDFT 的几种快速算法[71]。这些快速实 Fouirer 变换(FRFT)算法包括基 2 按时间抽取 FRFT、基 4 按时间抽取 FRFT、分离基按时间抽取 FRFT、素因子 FRFT 和 Winograd FRFT 算法。1991 年 Hu 和 Ersoy 又给出了计算任意点数 RDFT 的快速算法[72]。我们提出了利用实数 LMS 自适应算法计算 RDFT 与 DFT 的方法，运算量只有 Widrow 算法的$\frac{1}{3}$～$\frac{1}{5}$。文献[202-210，214-227]可供进一步研究 DFT 和 RDFT 快速算法时参阅。

## 1.2.2 DCT 和 DST 快速算法的发展

1. DCT **快速算法的发展**

DCT 的快速算法依据其计算途径可大致分为间接计算法和变换矩阵的直接因式分解法两类。前者通常采用快速 Fourier 变换或快速 Hartley 变换(FHT)间接求得 DCT，或将 DCT 转换成卷积来计算；后者利用四类 DCT 之间的关系，通过对 DCT 矩阵直接进行因式分解，达到提高计算速度的目的。

1974 年 Ahemd 最初引入 DCT 时指出可用一个 $2N$ 点 FFT 计算 $N$ 点 DCT[5]。1976 年 Haralick 改为用两个 $N$ 点 FFT 求得 $N$ 点 DCT[73]。接着 Narasimba 和 Peterson 以及 Markhoul 介绍了一种通过一重排序列的 $N$ 点 FFT 实现 $N$ 点 DCT 的方法[74,75]。Vetterli 和 Nussbaumer 也引用了这一方法[49]。1986 年 Malvar 证明了可通过同样长度的 DHT (DWT-Ⅰ)快速算法计算 DCT-Ⅱ[76]。1994 年茅一民给出利用二维 DHT 计算二维 DCT-Ⅱ的方法[155]。随后，Murthy 和 Sway 利用奇平方 DFT($O^2$DFT)快速算法计算 DCT-Ⅳ[228]。朱跃生[156]和我们分别提出了利用 DWT-Ⅲ与 DWT-Ⅱ快速算法计算 DCT-Ⅳ的方法。1994 年，Kumar 和 Prabhu 又提出了利用 DWT-Ⅰ(DHT)快速算法计算 DCT-Ⅰ，Ⅱ，Ⅲ，Ⅳ的方法[77]。1992 年 1 月 Heideman 证明了奇长度 DCT-Ⅱ可利用同长度 DFT 算法经简单换序求得[78]。同年 3 月，Chan 和 Ho 也发表了类似算法[79]，不同的是考虑了 DCT-Ⅱ，Ⅲ，Ⅳ三种类型。此外，Duhamel 和 Li 分别提出了将 DCT 转换成循环卷积或斜循环卷积进行计算的方法[157,158]。Chan 和 Siu 则提出用相关实现素长度 DCT[159]。

DCT 的第一个直接因式分解法是由 Chen 等人于 1977 年提出的[80]。尽管论文有一些错，但他们的方法却对正弦类变换快速算法的研究产生了重大的影响。1980 年 Wagh 和 Ganesh 利用群的概念将 DCT 矩阵划分成由子群组成的子矩阵，得到一种能灵活选择长度 $N$ 的 DCT 算法[81]，但算法较复杂且需较多数学知识。1984 年 B. G. Lee 提出一种使用正割乘子的 IDCT(DCT-Ⅲ)快速算法[82]。结构简单，很受关注。次年即被 Haque 推广到二维[83]。

Yip 和 Rao 也将 Lee 算法推广到其它正弦类变换[229,230]。1984-1986 年，Vetterli 和 Nussbaumer[49]、王中德[23,50]以及 Suehiro 等[51]又分别对 DCT 等正弦类变换算法进行改进。这几种算法均用正余弦作乘子。与 Lee 算法相比，计算量相同，但结构较复杂。1986 年 Hou 提出了用余弦作乘子的 DCT 递归算法[84]，之后有类似方法出现。时隔五年，Chan 和 Wu 等人同时分别将 Hou 算法通过矢量基的途径推广到二维[85,86]。1994 年 Britanak 给出 Hou 算法的广义信号流图及计算机程序[87]。1992 年 Cvetkovic 和 Popovic 又给出了 DCT-Ⅱ的正割乘子算法[88]。与 Lee 算法和 Hou 算法完全不同的是，我们 1992 年提出了分离基快速 DCT 算法。Feig 于 1990 年提出了一种分度 DCT 算法[89]。后来 Feig 和 Winograd 利用张量乘积推广到二维[90]。1995 年 Feig 和 Linzer 又构造了复合长度的分度 DCT 快速算法并推广到二维[160]。二维 DCT 还有另外两类算法，一类是将二维 DCT 转换成一维 DCT；另一类是转换为一维奇 DFT(ODFT)。转换方法有多项式变换和切比雪夫多项式变换两种。不同的转换方法和不同的一维算法得到不同的二维 DCT 算法[91-93]。Cho 和 S. U. Lee 利用两余弦函数积化和差公式，将 $N\times N$ 二维 DCT 转化成 $N$ 个一维 $N$ 点 DCT 来计算[161-163]。上述算法大都是针对长度 $N=2^m$ 点 DCT 设计。对于 $N\neq 2^m$ 的 DCT 算法，也有报道。1985 年和 1989 年 B. G. Lee 和 Yang 先后给出了 DCT 的素因子分解算法[94,95]。1994 年 P. Lee 提出了一种有效的 DCT 素因子算法[96]。1992 年 Steidl 根据切比雪夫多项式特性，导出复合长度 DCT 快速算法[97]。1993 年 5 月，我们提出了一种基 3DCT 快速算法。同年 11 月，Chan 和 Siu 发表了关于混合基 DCT 算法(基 3 和基 6 算法)的文章[98]，其中的基 3 算法与我们已提出的算法基本相同。1994 年 Chau 和 Wang 分别给出了用二阶递归滤波器计算任意长度 DCT 的方法[99,164]。1995 年 Chau 和 Siu 提出了把素长度 DCT 转换成递归滤波结构的新方法[165]。

由于低频 DCT 分量反映了信号的最有用信息，因此实际应用中只需计算低频 DCT 分量。据此，1991 年 Wang 提出了快速 DCT 的修枝算法[166]。1994 年 Skodras 提出的快速 DCT 修枝的新算法进一步减少了加法次数[167]。

还要一提的是 Yao 和 Hsu 以及 Yun 和 Lee 近几年对一些快速 DCT/IDCT 算法的有限字长误差分析进行了研究[168-171]。

2. DST **快速算法的发展**

DST 的快速算法也可分为间接计算法和直接分解法两类。Kekre 和 Solanki 1978 年引入 DST-Ⅰ时采用两个 $N$ 点 FFT 求得 $N$ 点 DST-Ⅰ[8]。1982 年王中德建立了 DST-Ⅱ与 DCT-Ⅱ之间的关系，给出了利用 DCT-Ⅱ快速算法求得 DST-Ⅱ的方法[100]。我们在 1992 年提出了利用 FHT 计算 DST-Ⅱ，Ⅲ的方法。1994 年 Kumar 和 Prabhu 在发表的用 FHT 计算 DST-Ⅰ，Ⅱ，Ⅲ，Ⅳ论文中也有类似算法[101]。朱跃生和我们还分别提出了利用 DWT-Ⅲ[156]与 DWT-Ⅱ快速算法计算 DST-Ⅳ的方法。1995 年 Tatsaki 提出了奇长度或偶长度的一维素因子 DST-Ⅱ算法，该算法用二维 DFT 间接计算 DST[172]。

DST 的直接分解法大多仿照 DCT 算法导出。1980 年 Yin 和 Rao 曾提出 DST-Ⅰ的稀疏矩阵分解法[102]，但论文中错误较多，无法实现[103]。Yip 和 Rao 将 Lee 提出的 DCT-Ⅲ算法推广到正弦变换的四种类型[229,230]。1992 年 Yao 和 Hsu 重复了这一工作，仿照 Lee 的 DCT-Ⅲ算法，给出 DST-Ⅲ的正割乘子算法[88]。同年 Cvetkovic 和 Popovic 又给出了 DST-Ⅱ的正割乘子算法[88]。1990 年王中德提出一种 DST-Ⅳ的快速算法[106]，由此可求得四种类型 DST。同年，Gupta 和 Rao 仿照 DCT 的 Hou 算法，发表了一种错误的 DST-Ⅱ快速递归算

法[107]。正确的算法我们已经提出,并且我们还提出了一种新的一维 DST-Ⅱ的快速递归算法,并已通过矢量基方式推广到二维。我们发表该算法的时间比 P. Lee 和 Huang1994 年在 IEEE Trans. SP 上发表早三个月[187]。文献[173-187]可供进一步研究 DCT/DST 快速算法时参阅。

### 1.2.3 DHT 或 DWT 快速算法的发展

1984 年 Bracewell 提出了按时间抽取的基 2FHT 算法[22]。次年 Meckelburg 和 Lipka 给出了按频率抽取的基 2FHT 算法[108]。随后引出了基 2 算法的一系列改进方法[109-111]。同一时期,王中德研究了四种 DWT 及其它正弦类变换的对称分解法[23]。随后王中德和 Suehiro 用不同的方案作了改进[50,51]。1985 年 Sorenson 等人用 Fortran 程序实现了基 2、基 4 以及按频率抽取分离基算法[112],次年 Pei 和 Wu 也发表了按频率抽取分离基算法的文章[113]。以后类似算法的文章又有发表。1990 年 Anupindi 和 Pradhu 提出了实对称序列的按频率抽取分离基算法[146]。1987 年 Hou 给出基 2 算法的矩阵分解形式[114]。而 Hsu 和 Wu 则建议用快速 Walsh-Hadamard 变换计算 DHT[115,116]。同年,王中德提出了快速 W 变换的正割乘子算法[117,118]。该算法结构简单,计算量小[147]。与此同时,吴建田将子群卷积算法移植到 DHT[119],郑宝玉随后推广到四种 DWT[120]。为了避免快速 W 变换使用正割乘子,1989 年提出的旋转因子 W 变换算法却不如正割乘子快速 W 变换算法的结构来得简单[154]。1988 年 Malvar 仿照 FFCT 给出的算法[121]和 1991 年 Chan 和 Siu 提出的将 $N$ 点 DHT 分解为一个 $N/2$ 点 DHT 与 $\frac{N}{4}$ 点对称余弦变换结构的算法[122],所需乘法次数与 DHT 分离基算法相同,但算法复杂些,加法次数也多些。1992 年 Yatsimirskiy 提出的 Rader-Brenner FHT 算法[123]也存在这一问题。目前,计算 DHT 或 DWT 所需计算量最少的分解算法,只有两种。一种是上面提到的正割乘子算法[117,118],另一种由 Duhamel 和 Vetterli 提出[124],将 $N$ 点 DHT 分解成一个 $\frac{N}{2}$ 点 DHT 和两个 $\frac{N}{4}$ 点 DFT 来计算。但该算法需借助 SRFFT,没有正割乘子算法结构简单。1992 年 Hu 引用 Duhamel-Vetterli 算法[124]计算广义 DHT,也就是 DWT-Ⅱ,Ⅲ,Ⅳ[125]。1997 年 Bi 提出了一种新的按频率抽取分离基算法[126],即其奇分量按 $X(4k+1)$ 和 $X(4k-1)$ 分解。这种方法的时间抽取形式我们在四年前就已发表。而我们提出的长度 $N=2^m$ 的分离基快速 W 变换(—Ⅱ,Ⅲ)算法,可计算四类 DWT,优于上面提到的任何算法。

至于 $N\neq 2^m$ 的 DHT 快速算法从 1990 年开始有些报道。1990 年 Yang 给出计算 DHT 的素因子算法[148]。为了避免实现素因子映射所需的额外运算,1992 年 Lun 和 Siu 提出了一种新的素因子映射[127],文中给出了 Fortran 子程序。1993 年 Meher 提出了一种新的素因子算法,用短序列 DHT 算法计算长序列 DHT[149]。1990 年茅一民将长度 $2^m$ 的 DHT 分离基算法推广到长度 $p^m$[128]。同年,Anupindi 等人提出基 3FHT 新算法[129]。我们和 Zhao[130]对此进行了改进,提出了原位基 3FHT 算法。我们的收稿日期比 Zhao 提前半年左右。我们还提出了基 3 快速 W 变换(—Ⅱ,Ⅲ)算法。1993 年 Lun 和 Siu 给出快速基 3/9DHT 算法[131]。1995 年 Bi 提出的基 3/9DHT 算法所需的乘法与 Lun 和 Siu 的算法相同,而减少了加法次数[132]。我们提出的基 3/9FHT 新算法与 Lun-Siu 算法、Bi 算法的实乘次数相同,而加法次数比这两种算法都少。

针对实际中仅有几个值不为零的情况,1991 年和 1996 年 Narayanan 和 Prabhu 分别提

出了基2按频率抽取FHT修枝法[150,151]。我们提出的变换分解法优于FHT修枝法。

二维DHT的变换核是不可分离的。可先进行行和列的奇偶分解，使分解后的变换核成为可分离的DCT和DST来加以计算[152,153]。Bracewell则通过先计算一种变换核可分离的中间变换[144]来计算二维DHT[133]。Kumaresan和Gupta提出了二维DHT的矢量基算法[134]。从1989年到1993年间，前后有多篇论文论述了二维DHT的分离矢量基算法[135-140]。或者将$N\times N$二维DHT分解成三个$\frac{N}{2}\times\frac{N}{2}$二维DHT和四个$\frac{N}{4}\times\frac{N}{4}$二维DHT[135,136]，或者将$N\times N$二维DHT分解成一个$\frac{N}{2}\times\frac{N}{2}$二维$DHT$和十二个$\frac{N}{4}\times\frac{N}{4}$二维DHT[137-140]。1990年，Chan和Ho提出了基于多项式变换的FHT算法[141]，该算法借助多项式变换把二维DHT映射成几个一维DWT-Ⅲ来计算。国内也有类似文章报道。Ma还将此方法推广到$p^n\times p^n$点DHT[142]。1997年曾泳泓将Bracewell在1986年提出的二维DHT算法推广到DWT-Ⅱ，Ⅲ，Ⅳ[143]，将二维DWT转化为可分离核的二维变换，然后用一维DWT计算二维DWT。1991年茅一民将Perkins提出的可分离Hartley式cas-cas变换[144]推广到四种类型，并给出了矢量基和分离矢量基算法[145]。

## 1.3 论文的内容安排

本文研究信号处理中的快速离散变换算法。一般按算法研究的国际通行标准，以实乘次数和实加次数作为衡量算法性能的主要指标。

第一章阐明了信号处理中一系列快速离散变换的提出过程，展示了这些快速离散变换在信号处理其它方面的应用，依据大量的有关文献，分门别类较全面地综述了离散Fourier变换与实离散Fourier变换、离散余弦变换与离散正弦变换、离散Hartley变换与离散W变换这三类离散变换快速算法的发展概况。

第二章研究有关离散Fourier变换与实离散Fourier变换的快速算法。提出以下五种新算法：(1)一种新的实乘子FFT-j算法；(2)ODFT的复共轭乘子分离基快速算法；(3)二维Fourier变换的一种高精度快速算法；(4)基于FFT的一维与二维信号快速内插算法；(5)利用实数LMS自适应滤波器计算RDFT与DFT。

第三章研究有关离散Hartley变换与离散W变换的快速算法。提出以下五种新算法：(1)快速Hartley变换的新算法；(2)分离基快速W变换(-Ⅱ，Ⅲ)算法；(3)基3FFT与FHT的新算法；(4)基3/9快速Hartley变换的新算法；(5)基于FHT的离散信号快速内插算法。

第四章研究有关离散余弦变换与离散正弦变换的快速算法。提出以下五种新算法：(1)分离基快速余弦变换算法；(2)利用快速W变换计算离散余弦变换-Ⅳ的新算法；(3)离散正弦变换-Ⅱ的快速递归算法；(4)二维矢量基离散正弦变换的快速算法；(5)基于快速W变换-Ⅱ计算离散正弦变换-Ⅳ的新算法。

第五章研究有关二维图像矩的快速算法。提出两种新算法。一种是计算二维图像矩的神经网络方法；另一种是计算图像矩的高精度快速算法。

第六章结束语。列出了本文研究工作的主要贡献，指出可进一步研究的若干问题。

# 第二章 离散 Fourier 变换与实离散 Fourier 变换的快速算法

## 2.1 一种新的实乘子 FFT-j 快速算法

本节引入了 DFT 的四种类型(即 DFT-j,j=Ⅰ,Ⅱ,Ⅲ,Ⅳ),探讨了四种 DFT 之间的联系以及其固有特性,由此提出了适合于长度 $N=2^m$ 所有四种类型 DFT 的一种新的实乘子快速算法。文中导出了用 Kronecker 直积与直和形式表示的算法公式;以长度 $N=2^3$ 为算例给出了信号流图;分析了算法的运算量,并与传统的基 2 算法以及现有的其它三种实乘子 FFT 算法进行了比较。结果表明,本节提出的算法所需运算量最少,用实数作乘子,可进行原位计算,结构简单规则,易于实现。

### 2.1.1 引 言

自从 1965 年 Cooley 和 Tukey 提出快速 Fourier 变换(FFT)算法后,离散 Fourier 变换(DFT)在众多领域得到极为广泛的应用。多年来人们不断探索,相继提出了一些新的 FFT 算法,以进一步降低运算量,减少存贮单元数目,扩展可适用范围,同时兼有结构的简单性与规则性,且易于实现。

由于预先计算实乘子比预先计算复乘子所需工作量减少一半,且存放实乘子所需存贮单元的数目只有存放复乘子所需存贮单元数目的一半,当处理大型数据时这种节省尤为可观。因此,研究实乘子 FFT 算法很有意义。现有的实乘子 FFT 算法主要有 Rader-Brenner 算法[44]、Cho-Temes 算法[45]以及 Preuss 算法[46]。与传统的基 2 算法相比,这些算法大大降低了所需的乘法次数,然而却以增加加法次数为代价,而且运算结构较为复杂,也不便于实现。

本节通过引入 DFT 的四种类型,依据四种 DFT 之间的联系以及其固有特性,提出了一种新的实乘子 FFT-j 算法。新算法适合于计算长度 $N=2^m$ 所有四种类型 DFT,所需运算量最少,用实数作乘子,可进行原位计算,结构简单规则,易于实现。

### 2.1.2 DFT 的四种类型

由于目前 DCT、DST 以及 DWT 各自均有四种类型[23,24],因此我们相应定义 DFT 的四种类型,记作 DFT-j,j=Ⅰ,Ⅱ,Ⅲ,Ⅳ。DFT-j 是一种将给定的长度为 $N$ 的输入复序列 $\{x(n);n=0,1,\cdots,N-1\}$ 变换为输出变换域复序列 $\{X_j(k);k=0,1,\cdots,N-1\}$ 的酉变换。其正反变换可用公式分别表示为

正变换($DFT$-$j$):

$$X_{\mathrm{I}}(k)=\mathrm{DFT\text{-}I}\left[x(n)\right]=\sum_{n=0}^{N-1}x(n)e^{-j\frac{2\pi}{N}kn} \tag{2.1}$$