一种 SAR 成像快速算法及优化实现*

苏 涛 庄德靖 吴顺君

(西安电子科技大学雷达信号处理重点实验室,西安 710071)

摘 要 针对实时 SAR 成像处理中相关运算量大的特点,提出了新的频域分解快速算法和优化设计方法. 根据参数优化准则,选择合适的填零个数和 DFT 点数分解方式,分析出利用频域分解和频域分段两种算法,可以大大减少相关运算中 FFT 变换的运算量. 在不同情况下,详细分析了多种算法的速度性能和适用性. 在并行多处理器上利用此算法,降低了 SAR 实时成像的运算量和成像延迟,显著提高了 SAR 实时成像的处理速度.

关键词 SAR 实时成像;分解算法;分段算法

中图分类号 TP75;TN958 **文献标识码** A

0 引言

合成孔径雷达(SAR)成像[1,2]与光学遥感成像相比[4,5],具有如下特点:全天候,不受白昼、气候影响;分辨力可达 0.5m,且分辨力与雷达、目标间的距离无关;有一定的穿透力,可探测地表下浅埋物体. SAR 技术近年来迅速成熟并实用化,已从军用推广到民用领域. 但 SAR 成像设备的成本较高,特别是 SAR 实时成像需要对宽带信号进行实时处理和较长时间的信号积累,所需要的运算速度很高、存储容量很大. 在采用数字信号处理技术进行成像时,需要进行大量的相关匹配运算. 虽然 FFT 变换可以大大降低相关匹配运算的运算量,但目前采用的算法仍然需要很大的设备量. 例如对一个 4000 * 4000 幅面成像所需的运算量高达 50 亿次,存储量高达 300 MB. 这对实时成像处理设备的运算能力、存储能力、体积、功耗都有严格要求. 本文针对 SAR 成像算法的特点,提出了一种快速算法,能显著降低对设备的要求.

1 SAR 成像的原理和实现方法

目前 SAR 成像多采用二维成像,其分辨力包括距离分辨力和方位分辨力. 在正侧视情况下(以下同),SAR 成像的距离向指与平台轨迹垂直的方向,方位向与平台轨迹一致. SAR 成像的距离分辨力是依靠发射大带宽的线性调频信号来获得的,在信号处理时要对接收回波进行脉冲压缩,这样达到了距离分辨力. 距离分辨力为 $\delta_r=\frac{c}{2B}$,c 是光速,B 是发射信号带宽,考虑到信号加权的影响,实际可获得的信号分辨力略微低于此式计算出的值. 信号带宽过高对发射信号产生、数模变换、运算速度、存储容量都提出很高要求. 例如 0.5 m 的距离分辨力意味着至少 300 MHz 信号带宽、300 MHz 的 AD 采样率.

SAR 成像的方位分辨力是依靠飞机、卫星等雷达载体平台在飞行时相对于目标的转动来获得的. SAR 成像的方位向理论分辨力 δ_{a0} 仅与波长 λ、波束宽度 θ 有关,$\delta_{a0}=\frac{D}{2}=\frac{\lambda}{2\theta}$,通常此分辨力比距离分辨力高得多. 实际中容易做到的分辨力为 $\delta_a=\frac{\lambda R_0}{2L}$,$L$ 是平台在相干积累时间内运行的路程,R_0 是雷达到目标的距离. 在相干积累时间内,雷达发射大量脉冲,不同方位的散射点,其回波的多普勒频率不同,以此来分辨方位上的各个点. 通常,实际能做到的方位分辨力要高于距离分辨力,即 $\delta_a<\delta_r$,为了做到距离、方位分辨力一致,方位向处理后的数据还要进行抽取.

图 1 是典型 SAR 成像的算法流程图. 与卷积、相关[3,6]一样,为了降低运算量,距离向处理时,在频域进行脉冲压缩和距离走动校正,方位向处理时,也是在频域进行方位压缩和相位校正,这些相关运算的实现都借助于快速傅里叶变换(FFT). 在SAR

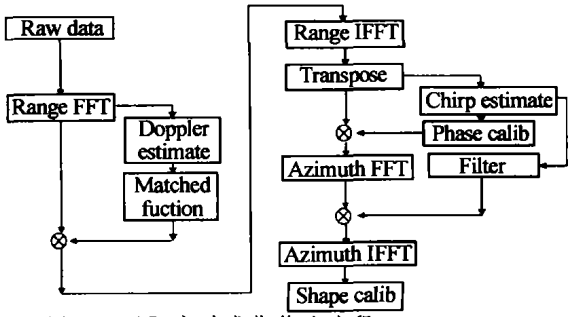


图 1 SAR 实时成像算法流程

Fig. 1 Flow chart of real time SAR imaging

*国防预研项目(41307010202)和总装跨行业基金(51407020201DZ0117)资助

Tel:029-88201581 Email:sutao@xidian.edu.cn

收稿日期:2004-07-26

成像信号处理的整个算法中，超过 80% 的运算是 FFT.

普通的基 2 或基 4FFT 算法要求 FFT 变换点数 N 必须是 2 的整数幂 2^m. 但是在 SAR 成像处理中，无论是距离向，还是方位向，处理的数据个数一般不能恰好满足该条件．现有算法采用普通添零方式，以凑够 2^m 个数据．对小于且接近于 2^m 的点数，补很少的零就可以凑成 2^m 长度，再进行 FFT 运算，这种情况下，添零对其运算效率不会有太大影响．但对于大于 2^{m-1} 且和 2^m 相差很大的点数，补的零就会很多，这样，运算量增大很多，需要的硬件设备量增大，处理效率降低．另外，当数据样点数远大于相关匹配滤波器系数个数时，采用分段滤波的算法也可以降低运算量．

2　快速相关算法

2.1　添零—分解 FFT 算法

由离散傅里叶变换(DFT)原理可知，若 $N\neq 2^m$ 但可分解，$N=N_1N_2$，长度为 N 的有限长序列的 DFT 可以分解为三个步骤：N_2 次 N_1 点 DFT，N_1N_2 次复乘，N_1 次 N_2 点 DFT. 因为 DFT 的运算量为 $O(N^2)$，分解后，三个步骤的总运算量明显小于未分解 DFT 的运算量．更进一步，若 $N_1\gg N_2$，$N_1=2^m$，则可以利用标准 FFT 算法实现 N_1 点的 DFT，从而快速计算出整个序列的 DFT. 这就是分解 DFT 算法．

当序列的点数N不可分解，或分解后的因子 $N_1\neq 2^m$ 时，可以对序列末尾添零凑成 N'，使得 N' 可分解，即如下关系成立

$$N\leqslant N'\leqslant 2^{[\log_2 N]+1} \tag{1}$$

$$N'=N_1\cdot N_2=2^m\cdot N_2 \tag{2}$$

式(1)中[]表示取整．然后借助分解 DFT 算法以减少运算量．这就是添零—分解 FFT 算法．当 N' 较大时，N_1、N_2 的取值有很多种，不同的取值会造成不同的运算量和程序复杂度．根据具体的 N 值，可选择合适的 N_1、N_2，使运算量最小．若 N_2 可继续分解，即 $N'=N_1N_2N_3$，$N_1=2^m$，则应选择合适的 N_1、N_2、N_3，使总运算量最小．为此，需要比较各种 N'取值、分解方法的运算量．

2.2　添零—分解 FFT 算法的运算量

当 FFT 的点数 N 很大时，N 点 FFT 基 2 算法的复乘、复加次数分别近似为 $\frac{1}{2}N\log_2 N$、$N\log_2 N$，这里仍将旋转因子为 1 的乘法次数统计在内．当 DFT 点数很小时，应不再统计旋转因子为 1 的乘法次数，因为 N 点 DFT 的复乘明显小于 N^2，为$(N-1)^2$. 下面将复乘、复加次数换算为实数乘、加次数：将一次复乘等同于 4 次实乘和 2 次实加，一次复加等同于 2 次实加．

普通添零算法将数据长度增加到 $N'=2^{[\log_2 N]+1}$，然后进行 FFT. FFT 的实乘、实加总运算量为

$$5\cdot 2^{[\log_2 N]+1}([\log_2 N]+1)=5\cdot N'\log_2 N' \tag{3}$$

而 2.1 节中的添零—分解 FFT 三个步骤的实乘、实加总运算量

$$5N_1N_2\log_2 N_1+8N_1N_2^2-8N_1N_2+6N_1=(5\log_2 N_1+8N_2-8)+6N_1 \tag{4}$$

N_1N_2 恒定时，若 N_1 越大，N_2 越小，即 $N_1\gg N_2$，则总运算量越小．因 $N_1=2^m$，因此 N_2 应是奇数．例如 $N_1N_2=3072$ 时，令 $N_1=1024$、$N_2=3$ 的运算量要少于 $N_1=512$、$N_2=6$.

若式(4)中的 N_2 是奇数，但可继续分解时，即 $N_1=2^m$，$N=N_1N_2N_3$，仍可使用上述过程分解计算 N_2N_3 点 DFT，分三步：N_3 次 N_2 点 DFT，N_2N_3 次复乘，N_2 次 N_3 点 DFT. 其运算量又比直接计算 N_2N_3 点的 DFT 要少．从减少运算量的角度考虑，在式(1)、(2)的约束下，N'不会分解出更多奇数因子．这时，添零—分解 FFT 算法的实乘、实加总运算量为

$$5N'\log_2 N_1-16N'+(8N'+6N_1)(N_2+N_3) \tag{5}$$

2.3　分解算法的相关运算量优化

用频域算法进行相关时，如果输入数据个数为 M，相关滤波器系数阶数为 K，需要进行 FFT 的数据样点数应是 $N=K+M-1$，但点数 N 一般不会恰好等于 2^m. 普通方法是添零到 $N'=2^m$，再进行 N'点 FFT、频域乘积、逆 FFT. 这种方法称为普通算法，总运算量以实乘或实加次数统计，为

$$10N'\log_2 N'+6N' \tag{6}$$

采用添零—分解 FFT 算法，若 $N'=2^mN_2=N_1N_2$，且奇数 N_2 不可再分解，频域滤波的运算量为

$$10N_1N_2\log_2 N_1+16N_1N_2^2-10N_1N_2+12N_1=N'(10\log_2 N_1+16N_2-10)+12N_1 \tag{7}$$

若 N' 可多次分解，即 $N'=2^mN_2N_3=N_1N_2N_3$，则分解算法的运算量为

$$10N'\log_2 N_1-26N'+(16N'+12N_1)(N_2+N_3) \tag{8}$$

要考虑如何添零，使得频域滤波算法的总运算量最小．在表达式 $N'=2^mN_2=N_1N_2$ 或 $N'=2^mN_2N_3=N_1N_2N_3$ 中，希望 N_1 尽可能大，奇数 N_2 或 N_2、N_3 尽可能小(N'不会分解出更多奇数因子)．下面提出搜索 N'、N_2 或 N'、N_2、N_3 的准则，选择

使运算量最小的添零—分解方式．步骤是：1）取 m 的初值为 $m=[\log_2 N]$，[]表示取整；第一种方案是直接添零到 $N'=2^{m+1}$，即普通算法；2）添零，使 $N'=2^{m-1}N_2$，N_2 是满足 $N\leqslant N'\leqslant 2^{[\log_2 N]+1}$ 的奇数；若 N_2 可再分解，分解后的运算量更小：小点数 DFT 的运算量可能大于大点数可分解 DFT 的运算量，如 3×3 点分解 DFT 比 7 点 DFT 的运算量小．因此，对同一 m 值，需要考虑多个 N_2 取值方案：一个不可分解的最小奇数和若干可分解的较小奇数，诸如 3、5、7、、11、13、等；3）将 m 减 1，$m=m-1$，重复上述式(2)，直到 $m=2$ 时停止搜索；4）比较所有各种方案中，按照式(7)或(8)，哪一种的运算量最小，即确定了 N'．当多种方案的运算量相差不大时，采用复杂度较小的方案．

本文将这种新的快速相关算法简称为分解算法．

根据实际需要滤波的数据点数，按上述分解快速算法搜索最优方案．例如样点数 N 为 9000．按上述方法搜索后，可以确定运算量从小到大的排列的次序是：$N'=512\times3\times3$、$N'=1024\times5$、$N'=2048\times3$、$N'=8192$．$N'=512\times3\times3$ 的运算量明显最小，但算法复杂度最高．$N'=8192$ 相当于普通算法，即直接添零，使 $N'=2^{m+1}$．

2.4 频域分段算法

用频域算法进行相关运算时，要考虑这样的问题：由于 FFT 的运算量随着点数的增加而超线性增长，因此将一个长序列分成若干数据段，分别进行频域相关，总运算量可能比不分段方法的运算量少．如果输入数据个数为 M，滤波器系数长度为 K，将数据等分成若干段，每段长度为 $L=2^m$，需分成 $[\frac{M+K-1}{L-K}]+1$ 段，[]表示取整，总运算量为

$$([\frac{M+K-1}{L-K}]+1)\cdot(10L\cdot\log_2 L+6L) \quad (11)$$

要注意每段之间要重叠 K 个数据，每段处理后只能得到 $L-K$ 个有效数据，所以，分段长度又不能太小．为使运算量最小，应对不同的 $N=M+K-1$，搜索最佳的 L 值．搜索 L 可以采用遍历法，不过，为减少搜索次数，可以这样：因为当 M、K 给定时，式(11)存在一个关于 L 的极小值点，但式(11)关于 L 不连续．所以，可以把式(11)近似为连续函数，见式(12)，求式(12)对 L 的导数，令其为零，得到 L_{opt}，通常 $L_{opt}\neq 2^m$，分别取 $L=2^{[\log_2 L_{opt}]}$、$L=2^{[\log_2 L_{opt}]+1}$，比较哪个运算量最小，即确定了最佳的 L．此方法简称为分段算法．

$$(\frac{M+K-1}{L-K})\cdot(10L\cdot\log_2 L+6L) \quad (12)$$

频域不分解算法（即普通算法）的运算量由式(6)确定．频域分解算法的运算量前文已讨论过．

图 2 是数据样点数 N 从 2 变化到 8192 时，采用三种频域算法的运算量．为便于比较，横坐标是 $N=M+K-1$．因为普通算法和分解算法对滤波器阶数 K 不敏感，图中为了清晰简洁，统计普通算法和分解算法的运算量时，不考虑 K．但分段算法的运算量与阶数密切相关．

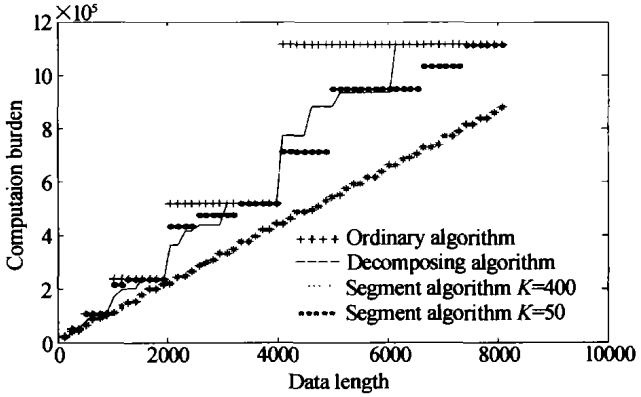


图 2 三种快速算法运算量与点数、阶数的关系

Fig. 2 Relationship between computation and length and taps of 3 fast algorithms

频域普通算法的运算量始终是最大的．以普通算法为基准，可以看到当 N 大于 256 时，分解算法在一半的情形下，即当 $2^{[\log_2 N]}<N\leqslant 1\frac{1}{2}\times 2^{[\log_2 N]}$ 时，可以显著降低运算量．但当 N 远大于滤波器阶数 K 时，采取分段卷积方法又比分解算法更有效．分段算法、分解算法的选择需要考虑 N、K 的比例．当 $N\gg K$ 时（约在 $\frac{N}{K}>10$ 时），分段算法优于分解算法．例如，$N=2100$，$K=50$ 时分段算法最好，$K=400$ 时分解算法最好．表 1 中以普通算法为标准，比较了其它算法的运算量与普通算法运算量的比值．

表 1 快速算法的运算量比较

N	1100		2200		3300		4400		5500	
K	50	400	50	400	50	400	50	400	50	400
分段算法	0.56	0.91	0.47	0.84	0.73	1	0.43	0.64	0.53	0.85
分解算法	0.72	0.72	0.70	0.70	1	1	0.69	0.69	0.84	0.84

各算法的复杂度从低到高依次为：频域普通算法、频域分段算法、频域分解算法．频域普通算法的适用范围最广泛．分段算法适用于连续信号或者信号长度远大于系数阶数的相关滤波．分解算法适用于非连续信号的相关滤波，其软件或硬件实现的复杂度比频域普通算法要高，用硬件来实现是困难的，

但利用现在流行的软件可编程 DSP 等处理器来实现是很方便的．

3 实现 SAR 实时成像

SAR 成像雷达的脉冲宽度 20 μs，信号带宽和采样频率为 80 MHz，则距离单元分辨力为 1.875 m，滤波器阶数 $K=1600$；条带宽度 5.5 km，对应距离单元数目为 $M=2933$ 点，数据样点数 $N=4533$ 点．方位分辨力需要的脉冲数目 4200. 参照文中方法，比较了三种频域算法的运算量，分解算法的运算量最小．在这里，分段算法之所以不如分解算法，是因为不满足 $N \gg K$ 的条件．算法性能如下：

距离向压缩，分解算法和普通算法的运算量之比为 0.69.

方位向压缩，分解算法和普通算法的运算量之比为 0.69.

综合考虑成像包含的其它处理，分解算法的总运算量只有普通算法的 75%. 即在相同的实时处理硬件平台上，分解算法可以比现有的普通算法的速度提高 25%.

在 8 片 TS101 组成的多处理器平台上，验证了上述算法．每片 TS101 的峰值运算能力为 18 亿/s，如果采用普通算法，距离和方位压缩都必须做 8192 点 FFT，距离压缩需要 4200 次 8192 点 FFT 和 4200 次 8192 点逆 FFT，方位压缩需要 2933 次 8192 点 FFT 和 2933 次 8192 点逆 FFT. 每个 8192 点 FFT 或逆 FFT 都需要 350 μs. 而分解算法只作相同次数的 4608 点 FFT，把 4608 点分解为 512×3×3 点，即 512 点 FFT 和 3 点 DFT，这样 4608 点 FFT 仅需要 245 μs. 成像速度由 1.2 幅/s 提高到 1.6 幅/s，速度提高了 30%，成像迟延也相应减少．同时，由于 FFT 点数的减少，需要的数据存储器容量也相应减少．

4 结论

本文针对实时 SAR 成像的特点，提出了一种填零—分解 FFT 变换的快速算法及其参数优化设计方法，在约 50%情况下可以减少频域相关处理的运算量．用此快速算法在多处理器上实现了 SAR 成像，显著提高了 SAR 实时成像的处理效率，降低了设备复杂度．此方法同样适用于系数阶数很大的其它频域匹配滤波应用．

参考文献

1 Xing M D. Properties of high-resolution range profiles. *Opt Eng*, 2002, **41**(2): 620～624

2 张直中．机载和星载合成孔径雷达导论．北京：电子工业出版社，2004. 5～20
Zhang Z Z. Principle of airborne and spaceborne synthetic aperture radar. Beijing: Electronic Industry Publish House, 2004. 5～20

3 冯迪，严瑛白，金国藩，等．求解分数傅里叶变换衍射积分的一种快速算法．光子学报，2003，**32**(7)：885～888
Feng D, Yan Y B, Jin G F, *et al*. *Acta Photonica Sinica*, 2003, **32**(7): 885～888

4 屈有山，田维坚，李英才，等．基于小波双三次插值提高光学遥感图像空间分辨率的研究．光子学报，2004，**33**(5)：601～604
Qu Y S, Tian W J, Li Y C, *et al*. *Acta Photonica Sinica*, 2004, **33**(5): 601～604

5 那彦，史林，杨万海．小波包变换与遥感图像融合．光子学报，2004，**33**(6)：736～738
Na Y, Shi L, Yang W H. *Acta Photonica Sinica*, 2004, **33**(6): 736～738

6 苏涛，吴顺君，李真芳．高性能 DSP 与高速实时信号处理．西安：西安电子科技大学出版社，2002. 225～243
Su T, Wu S J, Li Z F. High performance DSP and high speed real time signal processing. Xi′an: Xidian University Publish House: 2002. 225～243

Fast Algorithm for Real Time SAR Imaging and Implementation Optimizing

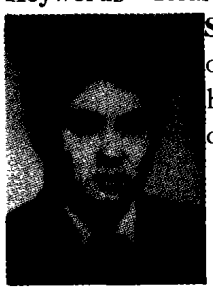
Su Tao, Zhuang Dejing, Wu Shunjun

Key Laboratory of Radar Signal Processing, Xidian University, Xi′an 710071

Received date: 2004-07-26

Abstract A novel fast algorithm of composition FFT for correlation processing in SAR imaging and its optimum scheme is presented. By choosing the number of padding zeros and decomposition scheme, the algorithm′s parameters are optimized, and the computational volume of FFT in correlation processing can be significantly lowered down. The speed performance and applicability of the algorithm is given in detail. It improves the speed and decreases the processing delay of real time SAR imaging on a parallel processor.

Keywords Real time SAR imaging; Decomposition algorithm; Segmenting algorithm



Su Tao Male, was born in 1968, received Ph. D. in 1999. He is now an associate professor of National Lab for Radar Signal Processing in Xidian University. He conducts research in high speed real time signal processing, fast algorithm and parallel processing system designing.