

doi:10.3969/j.issn.1001-893x.2013.06.020

基于 VSIPL 的跨平台算法中间件软件设计*

邓竹莎**

(中国西南电子技术研究所,成都 610036)

摘要:针对航空电子信息系统与通信数据链的嵌入式信息处理应用中算法软件移植性和可复用性差的问题,提出基于 VSIPL 标准应用编程接口的跨平台算法中间件软件设计模式,以增强多平台多系统软件可移植性及代码重用性;利用 Power-VSIPL 的 AltiVec 矢量加速、数据并行设计及循环展开,在保持原有算法处理精度基础上,提高软件运算效率 1~5 倍。

关键词:航空电子;信息系统;通信数据链;跨平台算法软件中间件;矢量信号图像处理标准库;矢量加速处理

中图分类号:TP311.1 文献标志码:A 文章编号:1001-893X(2013)06-0773-04

Design of Cross-platform Algorithms Middleware Based on VSIPL

DENG Zhu-sha

(Southwest China Institute of Electronic Technology, Chengdu 610036, China)

Abstract: A method based on VSIPL API is proposed to solve the problems such as bad portability and reusability of embedded information processing in aerospace and communication field to improve portability and reusability between various systems and platforms. By AltiVec technique of Power-VSIPL, data-parallel design and loop unrolling optimization, the middleware can insure precision and increase effectiveness by 1~5 times.

Key words: avionics; information system; communication datalink; cross-platform algorithm middleware; vector signal image processing library(VSIPL); vector acceleration

鉴于硬件设备的飞速发展,航空及通信领域嵌入式信息处理软件日益增多。由于与硬件绑定,此类软件模块定制性强,且缺乏统一的基础标准,在不同平台和环境中通用性差,重复开发情况严重,资金和人力资源大量浪费,技术延续性和质量稳定性难以保证。

此外,在嵌入式信息处理软件开发过程中,通常需要在 Matlab 环境仿真验证算法,再移植到嵌入式目标机运行。每次环境变化都会导致大量软件代码重新编写,而且一旦目标机研制进度拖延,后续软件开发和调试都会受到影响。

基于以上原因,本文针对航空和通信领域信息处理具体应用提取坐标转换、目标跟踪、状态估计及

图像处理算法基础模块,采用基于 VSIPL 的基础中间件^[1]开发算法功能软件包,通过底层硬件对矢量和矩阵运算的加速处理、数据并行化设计及循环展开等方式提高软件运行效率,为后续工程提供高效可靠的算法组件,使上层应用系统只需修改少量代码即可实现在不同平台的快速开发和移植。

1 VSIPL 中间件结构

VSIPL(Vector Signal Image Processing Library)是由美国国防部(DOD)下属的 DARPA 支持发起制定的标准应用编程接口,旨在建立一个嵌入式实时处理系统中可移植的高性能应用程序向量、信号和图

* 收稿日期:2013-02-07;修回日期:2013-04-23 Received date:2013-02-07;Revised date:2013-04-23

** 通讯作者:snoopyshylexp@yahoo.cn Corresponding author:snoopyshylexp@yahoo.cn

像处理的工业标准,从而最大程度地重用代码,大大缩短嵌入式领域项目工程的开发周期和维护时间。VSIPL 的软件结构如图 1 所示。

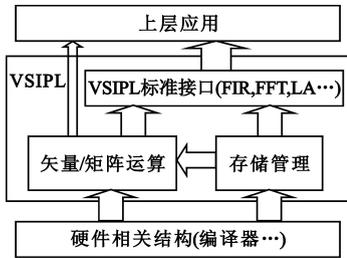


图 1 VSIPL 软件结构
Fig.1 VSIPL software architecture

这个 VSIPL API 程序库是目前嵌入式处理领域事实上的标准,主要用于军用级别的高端嵌入式信号、信息和图像处理。它通过为所有支持平台提供一致的 API 和一个公共的数学函数集合,可以方便地处理向量、矩阵、张量等对象,增强了应用程序的平台可移植性,同时通过代码重用,也加快了新技术的开发速度,大大缩短软硬件的开发周期。

目前硬件和软件供应商已经开发出许多 VSIPL 产品,作为开发高效率和可移植信号和图像处理应用计算程序的中间件,正被日益广泛地使用。

2 跨平台算法中间件软件设计

VSIPL 具有较好的跨平台特性和硬件加速特性,但作为一个基础中间件,缺少面向领域和应用的工具箱接口,不利于二次开发。本文在算法中间件设计过程中,一方面希望发挥 VSIPL 的标准化及硬件加速特性,另一方面旨在面向特定领域需求开发功能软件包,并针对数据特点进行优化,提升软件模块运行速度,提高可用性。

2.1 功能化组件模块设计

算法中间件设计时,首先需要划分软件功能。软件粒度根据需要可以是独立运行的子系统,也可以是程序模块。如果软件划分粒度太大,会造成功能复杂,升级困难,很难实现重用;如果粒度太小,又会造成组件数量太多,组件之间交互复杂,管理困难,性能低下。因此,功能化组件的划分粒度是软件设计的基础^[2-3]。

通常软件功能划分可分为系统、应用、业务组件

和模块 4 层,如图 2 所示。

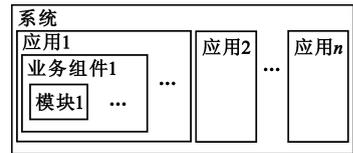


图 2 功能软件划分
Fig.2 Functional software partition

针对航空和通信等领域的具体需求及现有工程经验,本文主要生成模块级面向应用的基础算法组件(如滤波算法、图像分割算法等),供上层业务组件(如目标跟踪、状态估计等)调用,为后续工程应用开发提供可复用的软件模块。

拟开发的功能组件分为坐标转换、目标跟踪、状态估计、图像处理四大类,分别实现空间坐标转换、航迹滤波、航迹关联/融合、图像分割、图像压缩等具体功能,软件包划分及运行环境如图 3 所示。

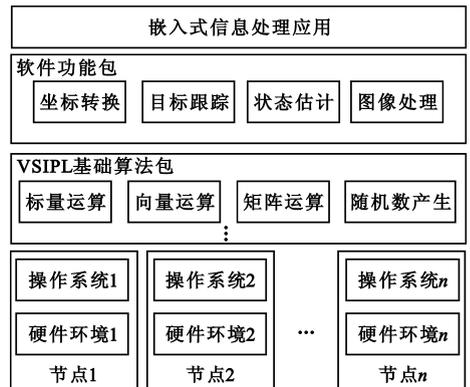


图 3 组件划分及运行环境
Fig.3 Component partition and running environment

2.2 跨平台软件设计

功能组件的跨平台设计通过各算法中间件一致的软件接口实现。针对目前嵌入式信息处理软件实际的开发流程,主要考虑 Matlab 仿真环境、PowerPC 刀片环境及 PowerPC 嵌入式环境下算法中间件软件设计。通过调用 Matlab-VSIPL、Power-VSIPL 和 Real-time-VSIPL 三部分基础组件,算法中间件屏蔽底层硬件平台及操作系统,为上层应用提供统一的标准 API,以确保不同平台不同系统间的无缝移植。

基于 VSIPL 的算法中间件开发流程如图 4 所示。其中,Matlab 环境算法中间件主要用于算法研究和仿真;PowerPC 刀片环境算法中间件实现模拟目标环境下的算法验证;PowerPC 嵌入式环境算法

中间件则针对嵌入式环境的实时处理要求,完成从算法仿真到目标机实现的最终移植。

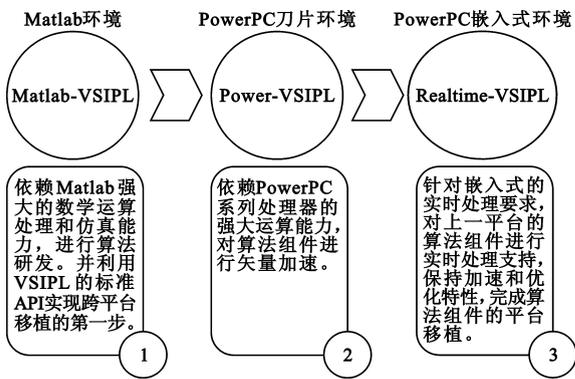


图 4 基于 VS IPL 的算法中间件开发流程
Fig.4 Development of algorithmic component based on VS IPL

算法中间件的基本结构如图 5 所示^[4-5]。

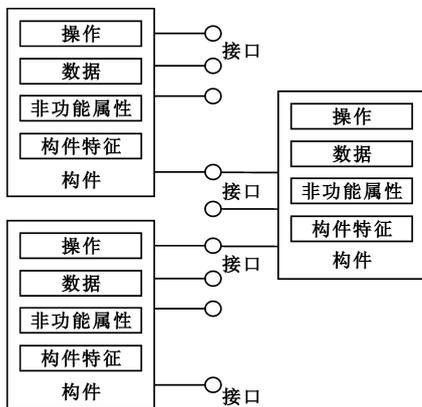


图 5 算法中间件结构
Fig.5 Architecture of algorithmic component

2.3 算法并行运算优化

算法中间件的优化处理,一方面依赖于 PowerPC 系列处理器自身的 Altivec 矢量加速技术,另一方面采用数据处理“并行化”^[6]的设计思路,充分发挥 VS IPL 基础库对矢量/矩阵运算的处理优势,将多点组成矢量或矩阵同时处理,缩短读取和存储数据的时间,增强运算能力。

以坐标转换为例,按照原有设计思路,坐标转换模块对接收到的传感器目标数据根据先后顺序实时排序,逐一处理,只有等第一个目标的坐标完成转换以后,第二个目标才会开始处理。原有处理流程如图 6 所示^[7-8],经过优化处理后,坐标转换模式如图 7 所示。

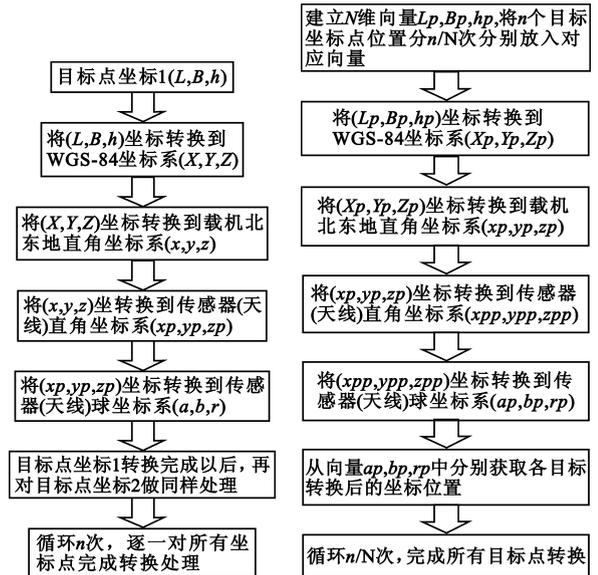


图 6 原有坐标转换流程
图 7 并行化的坐标转换处理流程
Fig.6 Old coordinate conversion
Fig.7 Parallel coordinate conversion

此外,算法模块(特别是状态估计)中循环遍历会占用大量计算时间,以目标关联为例,若 10 个传感器探测 1 000 个目标,为计算关联程度,极限情况下需完成 10^7 次运算,大大延迟了处理时间。本算法中间件设计时,通过循环展开,减小循环次数,充分发挥处理器本身流水线和指令预取技术的优势,达到软件加速的目的。

3 可移植性实现与运算性能分析

3.1 可移植性实现

算法中间件移植性实验选取 3 种代表性运行环境,从 Intel + Matlab 算法仿真平台将应用模块移植到 Linux + PowerPC 刀片环境,只需将 m 代码转换为对应的 C 代码,将其中调用的 Matlab-VS IPL 函数替换为对应的 Power-VS IPL 函数;从 Linux + PowerPC 刀片环境移植到 VxWorks + PowerPC 嵌入式目标机环境只需更换系统头文件,软件模块无需修改,大大缩短了开发周期。

3.2 运算性能分析

在 VxWorks + PowerPC 嵌入式目标机环境分别对目标位置坐标转换、目标跟踪、状态估计及图像处理模块进行性能测试,对各模块优化前后的处理时效进行比较。

实验一:将目标位置从经纬高转换到俯仰、方位、距离,再从俯仰、方位、距离转换到经纬高,分别对随机产生的 10 000、90 000、100 000 和 900 000 个点

计算平均处理时间,处理后的转换精度达到 10^{-7} , 计算时间见表 1。从测试结果可以看出,在保持原

有计算精度的前提下,新的坐标转换模块处理效率较过去提升 2~4 倍。

表 1 坐标转换测试结果

Table 1 Accelerated effect of coordinate convert arithmetic

并行计算一次 处理的点数	原有计算模式/ μs				优化后计算模式/ μs			
	10 000	90 000	100 000	900 000	10 000	90 000	100 000	900 000
100					1.666 67	1.666 67	1.500 00	1.611 11
1 000	3.333 33	3.888 89	4	4	1.666 67	1.296 30	1.333 33	1.259 26
5 000					1.666 67	1.296 30	1.333 33	1.240 74

实验二:目标作匀加速运动,对传感器探测到的目标量测数据进行滤波处理。输入 1 000 个量测值,优化前后跟踪算法耗时分别为 16 516 μs 和 9 183 μs , 处理效率提升约 1.8 倍。

实验三:采用最近邻算法对 4 个目标 600 组数据进行关联融合处理,优化前总处理时间为 4.2 s, 优化后处理时间 0.78 s, 处理效率提升约 5.4 倍。

实验四:采用分水岭图像分割算法对 256×256 图像进行分割,优化前后耗时分别为 27.956 6 s 和 18.743 3 s, 处理效率提升约 1.5 倍。

4 结 论

本文旨在采用 VSIPL 中间件,面向航空和通信领域具体需求开发算法中间件,并采用数据并行、循环展开等方式进行加速和优化,实现嵌入式信息处理软件在多平台多系统上的快速开发和移植。数值性能仿真说明,按照此模式设计的跨平台算法中间件可在保证计算精度的前提下大大提高处理时效,为信息处理嵌入式软件并行化设计与优化处理提供了新思路。

参考文献:

- [1] Janka R, Judd R, Lebak J, et al. VSIPL: an object - based open standard API for vector, signal, and image processing [C]//Proceedings of 2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Salt Lake City, UT: IEEE, 2001: 949 - 952.
- [2] 邹斌. 基于构件技术的嵌入式实时操作系统的研究 [D]. 南京:南京航空航天大学, 2010: 6 - 13.
ZOU Bin. The Research on Component-based Real-time Embedded Operation System [D]. Nanjing: Nanjing University of Aeronautics and Astronautics, 2010: 6 - 13. (in Chinese)
- [3] 华盈韬. 嵌入式 Linux 系统驱动构件的研究 [D]. 武汉: 华中科技大学, 2010: 8 - 11.
HUA Ying-tao. Research on Driver Component of Embedded

- Linux System [D]. Wuhan: Huazhong University of Science and Technology, 2010: 8 - 11. (in Chinese)
- [4] 戴鸿君. 基于异构多核体系与组件化软件的嵌入式系统研究 [D]. 浙江: 浙江大学, 2007: 70 - 74.
DAI Hong-jun. An Embedded System with the Architecture of Chip Multiprocessor and Component-Based Software [D]. Zhejiang: Zhejiang University, 2007: 70 - 74. (in Chinese)
- [5] 李超. 面向嵌入式领域的构件组装平台的设计与实现 [D]. 西安: 西安电子科技大学, 2011: 12 - 17.
LI Chao. Design and Implementation of Components Assembly Platform in Embedded Oriented Field [D]. Xi'an: Xi'an University of Electronic Technology, 2011: 12 - 17. (in Chinese)
- [6] 周伟明. 多核计算与程序设计 [M]. 武汉: 华中科技大学出版社, 2009: 316 - 322.
ZHOU Wei-ming. Multi-core Computing and Programming [M]. Wuhan: Huazhong University of Science and Technology Press, 2009: 316 - 322. (in Chinese)
- [7] 李锋, 金宏斌, 马建朝. 多雷达数据处理中坐标转换的新方法 [J]. 微计算机信息, 2007, 23(2 - 1): 303 - 305.
LI Feng, JIN Hong-bin, MA Jian-chao. A New Method of Coordinate Transformation Under Multi-Radar Data Processing System [J]. Microcomputer Information, February 2007, 23(2 - 1): 303 - 305. (in Chinese)
- [8] 廖鸣, 李鹏, 陈毅平, 等. 地心直角坐标和大地坐标转换算法研究 [J]. 电子科技, 2010, 23(2): 18 - 21.
LIAO Ming, LI Peng, CHEN Yi-ping, et al. Research on Transformation Methods from the Earth Rectangular Coordinate to the Geodetic Coordinate [J]. Electronic Science and Technology, 2010, 23(2): 18 - 21. (in Chinese)

作者简介:



邓竹莎 (1982—), 女, 四川康定人, 2006 年获硕士学位, 现为工程师, 主要研究方向为信息融合及嵌入式信息处理。

DENG Zhu-sha was born in Kangding, Sichuan Province, in 1982. She received the M.S. degree in 2006. She is now an engineer. Her research concerns information fusion and embedded information processing.

Email: snoopyshylexp@yahoo.cn