Implementing 3D SPHARM Surfaces Registration on Cell Processor

Huian Li
Pervasive Technology Institute
Indiana University
Indiana, USA
huili@iupui.edu

Mi Yan
IBM Systems Lab Services and Training
Indiana, USA
miyan@us.ibm.com

Robert Henschel
Pervasive Technology Institute
Indiana University
Indiana, USA
rhensche@indiana.edu

Li Shen
Dept. of Radiology
Indiana University School of Medicine
Indiana, USA
shenli@iupui.edu

***Abstract*—Spherical harmonics (SPHARM) description is a highly promising surface-based morphometry (SBM) method and has been widely used in neuroimaging applications to model the surface of arbitrarily shaped but simply connected 3D objects. This paper focuses on SHREC, a recently developed general-purpose surface-matching method for 3D SPHARM registration. We implemented SHREC in MATLAB, then optimized and implemented it on the Cell processor. Our experiments show that the Cell implementation on PowerXCell™ 8i is significantly faster than the MATLAB implementation on Intel Xeon E5335.**

Keywords-SPHARM Registraion; SHREC; CELL/B.E. processor; Load balance

I. INTRODUCTION

Surface-based morphometry (SBM) is an important research topic in brain imaging and other medical applications that aims to identify structural changes related to certain conditions, e.g., cortical thickness changes in autism [1]. Spherical harmonics (SPHARM) description [2] is a highly promising SBM method and has been widely used in neuroimaging applications to model the surface of arbitrarily shaped but simply connected 3D objects (e.g., hippocampus [3], brain cortex [1]). A crucial step for group analysis is shape registration that aims for establishing surface correspondence and aligning objects together. SHREC [3] is a recently developed general-purpose surface-matching method for registering 3D SPHARM models and is the focus of this paper.

SHREC, standing for ***S***P***H***ARM ***RE***gistration with I***C***P (iterative closest point [4]), uses a sampling-based approach for rotating the parameterization of an individual with different Euler angles and tries to find the best one that matches the parameterization of the template. This is a very time-consuming task, especially when a fine-resolution scheme is employed for sampling the rotation space. The kernel operation is the parameterization rotation of a SPHARM model.

In this work, we first implemented the SHREC algorithm in MATLAB, then optimized and implemented the algorithm onto the Cell processor.

II. SPHARM REGISTRATION

The SPHARM surface model of a 3D object can be represented based on a complete set of spherical harmonic basis functions Y_l^m, where Y_l^m denotes the spherical harmonic of degree l and order m. The SPHARM model can be represented in the following form:

$$v(\theta,\phi) = \sum_{l=0}^{L_{\max}} \sum_{m=-l}^{l} c_l^m Y_l^m(\theta,\phi) \tag{1}$$

where $c_l^m = (c_{xl}^m, c_{yl}^m, c_{zl}^m)^T$. The object surface can be reconstructed from SPHARM coefficients c_l^m.

The basic idea of SPHARM registration using SHREC is to fix one object and rotate the parameterization of the other one to find the position so that the surface distance between these two objects is minimized [3].

When an object is rotated by Euler angles (α,β,γ), a new set of SPHARM coefficients $c_l^m(\alpha\beta\gamma)$ can be calculated from the original coefficients c_l^m as follows [3]:

$$c_l^m(\alpha\beta\gamma) = \sum_{n=-l}^{l} D_{mn}^l(\alpha\beta\gamma) c_l^n \tag{2}$$

where

$$D_{mn}^l(\alpha\beta\gamma) = e^{(-i\alpha m - i\gamma n)} \left(\sum_{t=\max(0,n-m)}^{\min(l+n,l-m)} (-1)^t d_{mnt}^l(\beta) \right) \tag{3}$$

and

$$d^{l}_{mnt}(\beta) = \frac{\sqrt{(l+n)!(l-n)!(l+m)!(l-m)!}}{(l+n-t)!(l-m-t)!(t+m-n)!t!} \times (\cos\frac{\beta}{2})^{(2l+n-m-2t)} (\sin\frac{\beta}{2})^{(2t+m-n)} \quad (4)$$

The distance between two SPHARM models can be measured by root mean squared distance (RMSD). Let S_1 and S_2 be two SPHARM surfaces, and their SPHARM coefficients be $c^m_{1,l}$ and $c^m_{2,l}$, respectively. For $0 \le l \le L_{\max}$ and $-l \le m \le l$, the RMSD between S_1 and S_2 can be calculated as follows:

$$RMSD = \sqrt{\frac{1}{4\pi} \sum_{l=0}^{L_{\max}} \sum_{m=-l}^{l} \| c^m_{1,l} - c^m_{2,l} \|^2}$$

The rotation space $(\alpha\beta\gamma)$ can be sampled nearly uniformly using icosahedral subdivisions. With a straightforward MATLAB implementation, it took 823 seconds to do one rotation for harmonic degree 50 and it will take 3718 hours to complete the registration based on a level-3 icosahedral sampling scheme (i.e., 16267 samples in the rotation space) on HS21, a blade server with two 2GHz quad-core Intel Xeon E5335. The MATLAB code cannot complete SPHARM rotation for harmonic degree 85 or above due to an overflow in floating point computations.

III. Implementation of SPHARM registration on Cell processor

Cell/B.E. is a heterogeneous nine-core processor. Each Cell/B.E. has one 64-bit PowerPC Processing Element (PPE) and eight Synergistic Processing Elements (SPE). Each SPE has 128 128-bit registers and can execute 4-way single precision floating point instructions or 2-way double precision floating point instructions. Each SPE can directly access its own 256KB local store, but cannot access main memory directly. DMA transfers are required to move data between the main memory and the local store in SPEs.

We used two algorithmic optimization strategies to reduce the computation requirement in Equation (4). First, factorial is computed only once and the logarithmic results are kept in a lookup table *T* in order to avoid an overflow. Double precision is used to represent $T(i) = \log(i!)$ for $1 \le i \le 2L_{\max}$. Second, Equation (4) can be rewritten as below to replace expensive operations such as exponential with inexpensive operations such as multiplication:

$$\begin{aligned} d^{l}_{mnt}(\beta) &= \exp(\log(d^{l}_{mnt}(\beta)) \\ &= \exp(\\ &0.50 \times (T(l+n) + T(l-n) + T(l+m) + T(l-m)) \\ &- T(l+n-t) - T(l-m-t) - T(t+m-n) - T(t) \\ &+ (2l+n-m-2t) \times \log(\cos\frac{\beta}{2}) + (2t+m-n) \times \log(\sin\frac{\beta}{2})) \end{aligned} \quad (5)$$

For each rotation with Euler angle $(\alpha\beta\gamma)$, $\log(\cos\frac{\beta}{2})$ and $\log(\sin\frac{\beta}{2})$ only need to be computed once. The computation cost in Equation (5) is much less than in Equation (4).

The basic strategy to implement SPHAM registration on Cell/B.E. is to decompose the domain space $0 \le l \le L_{\max}$ and $-l \le m \le l$ onto multiple SPEs. Each SPE computes $c^m_l(\alpha\beta\gamma)$ for (l,m) and computes partial RMSD in its own sub-domain. The domain space is not rectangular. Simply decomposition along the l direction will lead to severe load imbalance. We may do decomposition along the m direction to obtain good balance, but the fine grain parallelism introduces synchronization and communication overhead between SPEs. We choose to use an alternative approach to distribute (l,m) onto SPEs. For each pair (l,m) in the domain space, we assign a unique index $f_{(l,m)} = l^2 + m + l$. Obviously, there is one to one mapping between (l,m) and f for $0 \le f \le L_{\max}{}^2 + 2L_{\max}$. Each SPE works on $(L_{\max}+1)^2 / s$ pairs of (l,m), assuming s SPEs are in use.

IV. Experimental results on PowerXCell™ 8i

We implemented SPHARM registration using SHREC with Cell SDK 3.1 and measured the performance on an IBM QS22, a Cell/B.E. blade with two 3.2GHz IBM PowerXCell™ 8i processors. Each PowerXCell™ 8i has a peak performance of 200 GFLOPS for single precision and 100 GFLOPS for double precision.

We used dual source compiler IBM XLC 10.1. The application was manually split into PPE code and SPE code. PPE code and SPE code are compiled with ppuxlc and spuxlc, respectively. In SHREC, three SPHARM models are needed to be kept in the local store on each SPE. The data can not fit into the local store, so DMA transfer between the main memory and the local store is involved. SPU decrementer cycle was used to measure the time spent on DMA transfer and on computation. Our test shows that, in SHREC, time on DMA transfer is insignificant comparing to the time on computation. So, we did not use asynchronous DMA and double buffer to hide DMA latency.

At first, we used single precision to compute $c^m_l(\alpha\beta\gamma)$, but the image reconstructed with the new SPHARM coefficients suffered from quality loss. In order to maintain reasonable image quality, double precision is necessary in SPHARM registration. In order to take advantage of 2-way double precision SIMD instructions, extra data rearrangement is needed to align the data buffer to 16 bytes at each iteration. Our experiment showed that 2-way SIMD implementation does not improve the performance significantly due to the extra data rearrangement.

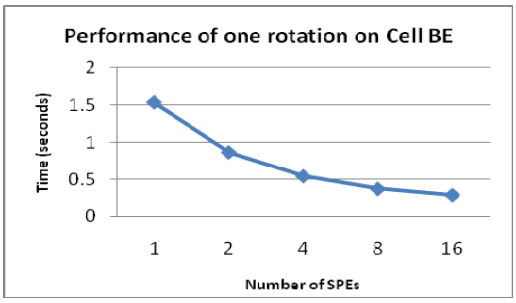

Figure 1. Performance of one rotation on Cell/BE

We did experiments on QS22 based on a level-3 icosahedral sampling scheme. The performance results are shown in figures 1 and 2. Figure 1 shows the running time of one SPHARM rotation. The speedups of Cell/B.E. implementation of one rotation on QS22 using 1, 2, 4, 8, and 16 SPEs against the MATLAB implementation on HS21 are 534, 943, 1505, 2200, and 2826, respectively. With the high performance implementation on Cell/B.E., the SHREC can be solved on QS22 in reasonable time. Figure 2 shows the running time to register two SPHARM models using SHREC and to find the shortest RMSD distance between them.

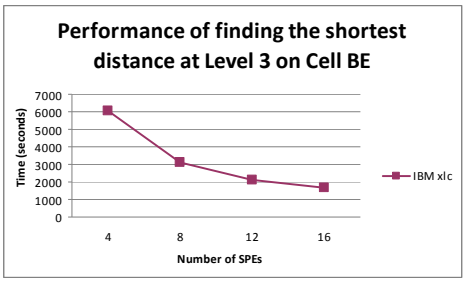

Figure 2. Performance of finding the shortest distance at Level 3 on Cell/BE

V. CONCLUSIONS

SPHARM registration with SHREC is a computation intensive algorithm. It takes months to solve this problem with MATLAB on HS21. However, with algorithmic optimization and careful implementation on Cell/B.E., it takes less than one hour on a single PowerXCell™ 8i processor.

REFERENCES

[1] M. K. Chung, K. M. Dalton, L. Shen, A. C. Evans, and R. J. Davidson, "Weighted Fourier Series Representation and Its Application to Quantifying Amount of Gray Matter," *IEEE Transactions on Medical Imaging, Special Issue on Computational Neuroanatomy (invited paper)*, vol. 26, pp. 566-581, 2007.

[2] C. Brechbuhler, G. Gerig, and O. Kubler, "Parametrization of closed surfaces for 3D shape description," *Computer Vision and Image Understanding*, vol. 61, pp. 154-170, 1995.

[3] L. Shen, H. Huang, F. Makdeon, and A. J. Saykin, "Efficient Registration of 3D SPHARM Surfaces," in *CRV 2007: Fourth Canadian Conference on Computer and Robot Vision*, Montreal, QC., 2007, pp. 81-88.

[4] P. J. Besl and N. D. McKay. "A Method for Registration of 3-D Shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, issue 2, pp. 239-256, 1992.