

GPU 実装による Jones matrix OCT 用 maximum a-posteriori 複屈折推定器 GPU based high-speed implementation of maximum a posteriori birefringence estimator for Jones matrix OCT

○久保田敦¹⁾, 安野嘉晃²⁾, 巻田修一²⁾

○Atsushi Kubota¹⁾, Yoshiaki Yasuno²⁾, Shuichi Makita²⁾

(株) スカイテクノロジー¹⁾, COG 筑波大学²⁾

Sky Technology Inc.¹⁾, Computational Optics Group, University of Tsukuba²⁾

E-mail: a_kubota@skytechnology.co.jp

The maximum-a-posteriori birefringence estimator is a highly sensitive birefringence estimation algorithm for polarization sensitive OCT. This can accurately estimate birefringence, but requires a long calculation time. The speed of the estimator was increased using GPU, and the speed was compared between several GPU units.

1. はじめに

Jones-matrix OCT などの偏光 OCT を用いて局所位相遅延量を計測することで試料の複屈折分布を知ることができることが知られている。しかし、通常、局所位相遅延量は小さく、そのため、複屈折計測の信号雑音比(SNR)は低い。この問題を解決するために事後分布最大化(MAP)複屈折推定器¹⁾が提案されている。この方法は複数の SNR の低い局所位相遅延の計測量から試料の複屈折値を最尤推定する。これは、数学的に厳密な方法であるが、長い計算時間を必要とする。従来の CPU による計算を用いた実装では、三次元分布の複屈折推定 40 分程かかっていた。

本研究では GPU を用いた MAP 複屈折推定器の実装を行い、複屈折推定の速度改善を行った。さらに、複数の GPU を用いて速度比較を行うことで、推定速度の制限要因を検討した。

2. 手法

MAP 複屈折推定器は「ある有効 SNR (ESNR) 値と複屈折値が計測された場合の真の複屈折の確率分布」(尤度関数)を利用して推定を行う。この尤度関数はモンテカルロ法によってすべての ESNR と複屈折地の組み合わせに対して予め計算され、三次元の配列データとしてメインメモリ内に保存されている。実際の推定過程では、試料の同一箇所を複数回計測し、その計測毎に一次元の尤度関数をその三次元配列から参照し、すべての計測の尤度関数を掛け合わせることで真の複屈折の事後分布を求め、それが最大値をとるような複屈折値を MAP 推定値とする。

GPU 演算は、アクセス速度が速いが小容量の shared memory (streaming multiprocessor 一つ当たり 96kB) を効率よく使用し、より多くの thread を同時実行することで高速化される。そこで、本研究では、画像上一ピクセルの推定処理を一つの thread で処理することで、処理の並列化・高速化を行った。

3. 結果

各点を 4 回ずつ繰り返し計測した三次元 JM-OCT データ (512 × 640 × 256 pixel) を用いて複屈折の MAP 推定を行った。以下に複数の GPU モデル、および従来の CPU を用いた実装による推定速度の結果を示す。

Table 1 The processing performance of MAP birefringence estimates and GPU specs.

Processor	Processing time/slice[sec]	Total processing time (256 slices) [sec]	Total # of parallelly processable thread	Total # of CUDA cores
GTX 1060	0.77	195.98	240	1280
GTX 1070M	0.47	119.44	384	2048
GTX 1080Ti	0.31	78368	672	3584
CPU※	6.30	1,637	(unknown)	---

※ Core i7-7700HQ 2.8GHz Processed with python 2.7 with Numpy1.16.4(OpenBlas)

CPU に比べ GPU では 8 倍の速度向上が実現された。GPU モデル間の比較から、処理速度とは並列実行可能な thread 数に対して線形に向上することがわかった。

4. 結論

GPU 実装を用いて、CPU よりも 8 倍以上光速な MAP 推定が実現された。また、GPU による MAP 推定の処理速度は並列実行可能な thread 数によって決定されていることがわかった。

参考文献 1) D. Kasaragod, S. Makita, Y.-J. Hong, and Y. Yasuno, *Biomed. Opt. Express* **8**, 653–669 (2017).