POSTER 03

GPU Based PRISM Calculation for High Resolution Estimation of Precipitation

J. H. Park[1*], Y. S. Shin[1], W. S. Kang[1], S. K. Kim[1], Y. K. Han[1], and E. W. Park[2]

[1]*R&D Center, EPINET Co., Ltd, Anyang 14056, Korea,*

[2]*Department of Agricultural Biotechnology, Seoul National University, Seoul 08826, Korea*

I. Introduction

This study is part of the research development project of the Rural Development Administration's Agenda. The purpose of this study is to enhance the speed of PRISM (Parameter-elevation Regression on Independent Slopes Model), which is applied to the Agricultural Climate and Disaster Early Warning System (agmet.kr) provided to sites around the Seomjin river. PRISM is a statistically downscaling technology of climate information drawn from geological information such as distance, altitude, coastal proximity, and topographic facet. This study used the PRISM method, which uses the local precipitation reports of the Korea Meteorological Administration (KMA) at 5km-spatial resolution as the virtual observation point to increase the spatial resolution to 270 m (Kim *et al.*, 2013), then attempted to increase its speed by performing data-parallel processing using GPU.

While the current PRISM precipitation estimation is made with high-performance CPU that has 8 physical cores, as it not only calculates estimations for several cities and districts (Hadong, Gurye, Gwangyang) simultaneously but needs to be able to accommodate a large number of cities and districts to be added in the future, speed improvement and system expansion are necessary. Therefore, by calculating precipitation with the GPU-based PRISM, maximum speed improvement could be achieved with minimum cost.

Upon a nationwide testing with 270 m-spatial resolution, we found that GPU performed at least 500 times faster on average than CPU, and such ratio increased in proportion to the subject grid or observation point. Therefore, even when the test region is expanded to nationwide and spatial resolution is enhanced, it is expected to produce PRISM precipitation in stable intervals by the second.

* Correspondence to : parkjh@epinet.kr

II. Materials and Methods

2.1. Possibility and method of GPU application on PRISM precipitation

The extent of speed improvement made by parallel processing of the model depends on the ratio of the portion within the model eligible for parallel processing (Amdahl *et al.*, 1967). If 20% of the entire given time is eligible for parallel processing, even if that portion is accelerated by 1,000 times, the total speed cannot be enhanced beyond a maximum of 1.25 times. On the other hand, if 99% is eligible, the maximum speed enhancement would be 91 times when this portion is accelerated by 1,000 times. All of the numerical calculations of the PRISM precipitation are eligible for parallel processing except for the reading and writing of the input and output files, respectively, and increased calculations lead to greater speed improvement results. (Rodgers *et al.*, 1985)

Each of the grid points of the PRISM precipitation can be processed independently of the other grids. (Fig. 1) Also, the input data are standard grid data that can be effectively run through GPU, and this could be used as the GPU's texture material. The grid closer to the grid point for estimation tends to more frequently referenced, and this acts as a high cache hit rate in the GPU, reducing the global memory reference and significantly contributing to speed improvement. In Fig. 1, red indicates the grid intended for estimation and the green indicates the grid with the observed value. The values of the red, green, and blue grids from the DEM and Aspect grid sets are used to estimate the value of the red grid, and the same is applied to all grids intended for estimation.

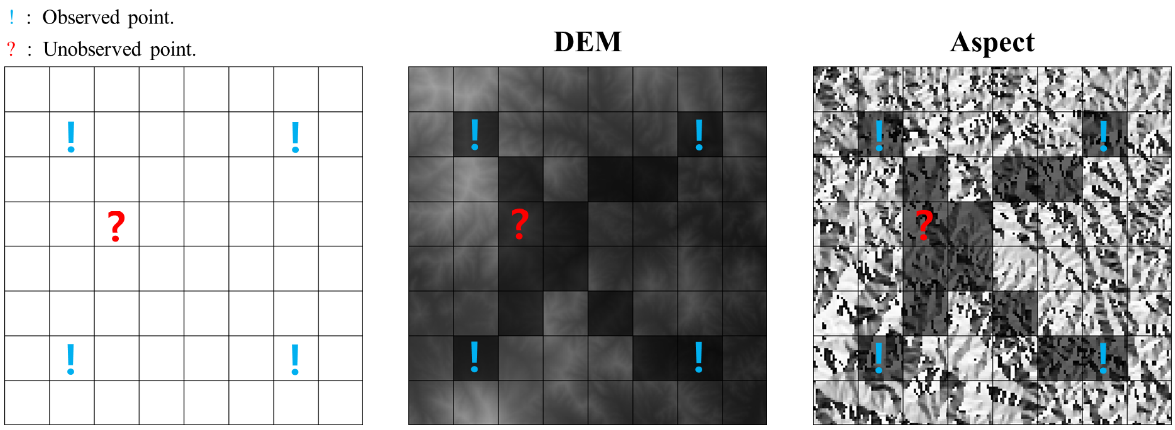

Fig. 1. Lattices used to estimate one lattice value.

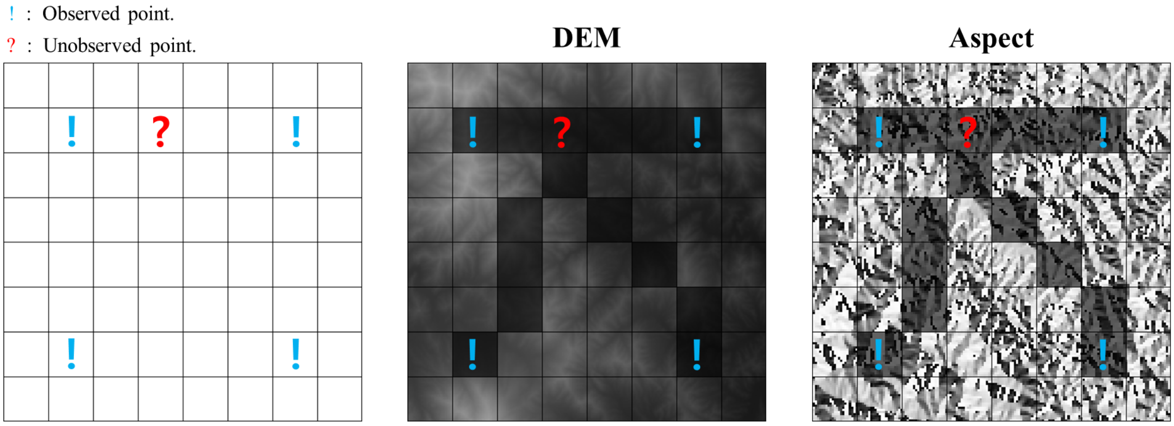

Fig. 2. Lattices used to estimate another lattice value.

2.2. Composition of the calculation system

The hardware and software specifications of the computing system used in this study are shown in Table 1. For the CPU, it was executed with Python and all 8 physical cores were used in the calculation. Also, in order to mitigate the time-lapse issue rising from linguistic characteristics, the task was conducted by using pypy. GPU was also executed with Python, and OpenCL was used for the parallel processing. Also, iterative partitioning with 256 × 256 tiles was performed for the high resolution data.

Table 1. CPU and GPU system specification

	CPU	GPU
Manufacturer	Intel	NVIDIA
Model	Xeon E5-2650 v2	GeForce GTX 660
RAM	64GB	1GB
lowest price	$1453.00	$119.99
Release Date	July 1, 2013	September 13, 2012
OS	Linux (Cent OS 7.0)	Windows 10

* lowest price: August 5, 2016 according to amazon.com.

III. Results

The CPU no cached method was most time consuming because the weight is calculated every time. On the other hand, the cached method, in which weight calculation is performed once then cached and reused, can be accelerated by approximately 10 times compared to the no cached method. However, it cannot be used if there is any alteration with the observation point (such as transfer, elimination, addition, etc.), coverage area, or resolution. The GPU processing method was able to improve speed by an average of 500 times without any particular restrictions as the ones mentioned above, and the calculation can be completed in a relatively stable timeframe even if the scope of the estimation is expanded.

This study did not consider cases of using input data other than DEM and Aspect. Therefore, moving forward, it would be necessary to analyze the degree of maintaining stability in other PRISMs and make improvements.

Table 2. Variation of Calculation Time According to Changes in the Area

Area (km^2)	Area Unit	GPU (sec)	CPU (sec)	
			cached	no cached
127	eup	0.55	2	13
2131	si	0.58	8	146
8203		0.74	43	367
10318		0.86	132	1,117
24569	do	1.47	242	2,152
55092		2.68	710	4,564
76408		3.08	1,101	8,831
88880		4.36	1,697	14,345
103607	South Korea	5.01	2,852	26,996

* Area: Calculation based on 270 m DEM.

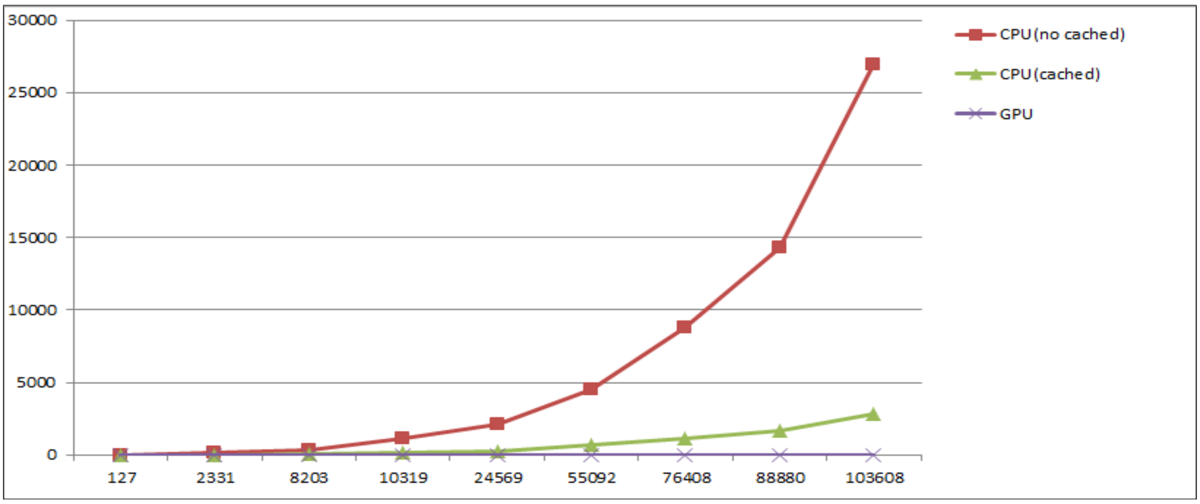

Fig. 3. Variation of Calculation Time According to Changes in the Area.

Acknowledgements

This study was conducted with the support of the Rural Development Administration’s initiative to “Develop an Agricultural Climate and Disaster Early Warning System to Address Extraordinary Climate Changes.” (Item No. PJ010007062016)

References

Amdahl, Gene M., 1967: Validity of the single processor approach to achieving large-scale computing capabilities. *AFIPS Conference Proceedings* **30**, 483–485.

Rodgers, David P., 1985: Improvements in multiprocessor system design. *ACM SIGARCH Computer Architecture News archive* **13**(3), 225–231

Chung, U., K. Yun, K. S. Cho, J. H. Yi, and J. I. Yun, 2009: The PRISM-based rainfall mapping at an enhanced gridcell resolution in complex terrain. *Korean Journal of Agricultural and Forest Meteorology* **11**(2), 72-78.

Kim, D. J., and J. I. Yun, 2013: Improving usage of the Korea meteorological administration's digital forecasts in agriculture: 2. Refining the distribution of precipitation amount. *Korean Journal of Agricultural and Forest Meteorology* **15**(3), 171-177.

Khronos Group, Open Computing Language. http://www.khronos.org/opencl/