Studies of Large-Scale Data Visualization, GPGPU Application and Visual Data Mining

Project Representative
Fumiaki Araki Earth Simulator Center, Japan Agency for Marine-Earth Science and Technology

Authors
Fumiaki Araki*1, Shintaro Kawahara*1, Nobuaki Ohno*1and Daisuke Matsuoka*1
*1 Earth Simulator Center, Japan Agency for Marine-Earth Science and Technology

Research and development for visualization technologies carried out by Advanced Visualization and Perception Research Group of the Earth Simulator Center in the fiscal year 2009 is reported. In terms of in-situ visualization, speeding up and vectorization of several pre-processing parts of a visualization program containing software-rendering algorithms is considered and examined. A ray-skip algorithm with octree-structured bounding volumes is implemented into pre-processing parts of the visualization program to accelerate ray-casting calculations. In terms of an application of GPGPU, implementation methods of GPGPU technology to a virtual reality visualization program, especially for an isosurface reconstruction algorithm, are examined. It is confirmed that the processing time including drawing was greatly shortened in comparison with CPU implementation. Implementations of it to each process of CAVELib program and memory managements are also discussed. In terms of visual data mining, our developed methods can classify magnetic field lines automatically into three types based on the difference between their topologies and detect the number of loop times. 3D distribution of magnetic field lines' topology and its separatrix surfaces are visualized by using this automatic classification method. Brief overviews of the survey for recent researches and developments in terms of large-scale data visualization are reported. Current progress of VFIVE development is also reported.

Keywords: large-scale data visualization, vectorization, GPGPU, virtual reality visualization, visual data mining

1. Introduction

Computer technology is quickly advancing and has become able to process more tremendously than before. Typical sizes of datasets produced by high-resolution simulations on recent supercomputers are enlarged from terabytes to petabytes. It would be a serious problem if scientists could not get any knowledge from the data because its size is too much to analyze it. To avoid this situation, it is important to develop fully-powerful and useful tools to visualize such the massive data at high speed and extract hidden information effectively. We are focusing on the problems on visualizing large-scale datasets and proceeding to study the visualization methodology and develop various useful tools.

In the fiscal year 2009, we have tried following six themes; vectorization of visualization algorithm, application of the general purpose computing on graphics processing units (GPGPU) with virtual reality visualization, visual data mining approach, a survey for large scale data visualization, and VFIVE development. For the first theme, one of the solutions to visualize massive data produced by a large-scale simulation is to execute the visualizing process at the same computational environment to do the simulation. In the case that the environment is a supercomputer system with vector processors like as the Earth Simulator, it is important to vectorize the visualization algorithm, in addition to parallelize it. We explore the possibility to do it in this section. For the second theme, GPGPU has nowadays become used for various areas that need to execute massive calculations. We think that applying GPGPU technology with speeding up visualization processes is important for interactive visualization methods, and especially our virtual reality (VR) visualization method, which has to keep both immediate response and interactivity to gain highly immersive sense. This is denoted in Section 3. For the third theme, we introduce an application example with our visual data mining approach to an analysis for the data of a space plasma simulation in Section 4. Magnetic field lines reproduced from that dataset are classified automatically on the basis of a few topologically different types. In Section 5, as the fourth theme, we describe brief overviews of our investigations of recent researches and developments in terms of large-scale data visualization, in order to take a comprehensive view of this research area and find clues toward to the next-generation visualization paradigm. In Section 6, progress of our virtual reality software VFIVE is reported. The last section is devoted to summery.

2. Vectorization of visualization algorithm

In the fiscal year 2009, we prepared a visualization program for software-rendering and vectorized the pre-processing parts of it. This program is implemented with both of isosurface and volume rendering functions based on the ray-casting algorithm for scalar computers without GPU. This program is also equipped with pre-processing parts to accelerate those ray-casting calculations. Each of these pre-processes acts as following; (1) to generate bounding volumes, (2) to memorize the empty regions of a target volume data to an octree data, and (3) to do ray-skip at each empty region based on this octree data, as shown in Fig. 1. Computation time for the ray-casting is greatly reduced through these pre-processing. Another pre-processing is also equipped into this visualization program. That is for derivation of gradient vectors of the target volume data which are used as the normal vectors to shade both of isosurface and volume rendered objects. Vectorization is applied with all of the pre-processing parts of this visualization program. We confirmed that computation time is successfully reduced by the vectorization on vector type computers. We are planning to vectorize the main part of this program, the ray-casting algorithm, of this software.

3. Application of GPGPU with virtual reality visualization

We tried to apply the general purpose computing on graphics processing units (GPGPU) [1] technology to the visualization using a virtual reality (VR) system. In the fiscal year 2009, we examined implementation methods of this technology to a visualization program for BRAVE [2], which is a CAVE system [3] of the Earth Simulator Center, and evaluated whether the GPGPU technology was effective for the acceleration of the program. Using high-speed calculation functions of GPGPU, we aimed to keep interactivity of visualization programs for VR system and enable to visualize a much larger volume of data than before.

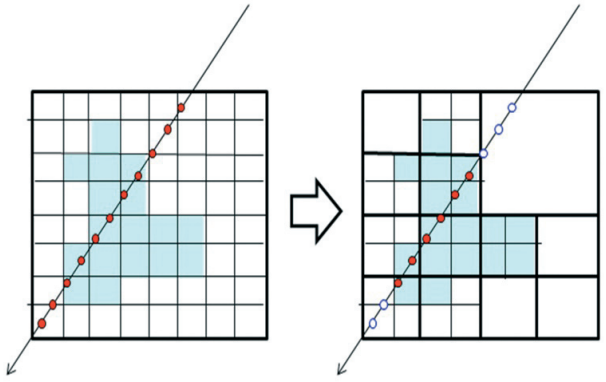

Fig. 1 Concept of bounding volume. When volume rendering is carried out by ray casting method, the time for calculation is greatly reduced by setting bounding volumes distinguishing from empty regions and non-empty regions in data and making ray skip the empty regions.

Specifically, we implemented the isosurface reconstruction algorithm that was written in GPU code to the visualization program for VR system and evaluated the performance. We developed the prototype program using CAVELib [4] as API for VR system, CUDA [5] as the GPGPU programming language, and OpenGL as 3-D CG API. As the isosurface reconstruction algorithm written in GPU code, we improved and used a sample code included in CUDA SDK. This sample code used Marching Cubes method as an isosurface reconstruction algorithm. In the performance assessment described later, we compared this implementation with VFIVE [2, 6, 7], because same algorithm (Marching Cubes) is also implemented to the isosurfacing function of VFIVE.

Fig. 2 is the hardware configuration of graphics workstation generating stereoscopic images to screens of BRAVE. Four graphics cards (NVIDIA Quadro FX 5600) are connected to graphics workstation (SGI Asterism: 8CPU/16Cores, 128GB Memory) with PCI-Express. Each graphics card is connected to the projector of BRAVE, and renders stereoscopic images. Fig. 3 is program behavior written by CAVELib. Processes executed in each thread are divided broadly into three parts: the initialization part, the calculation part and the rendering part. At the start of the program, four threads are launched corresponding to each screen of BRAVE. Then, calculation and rendering process are executed repeatedly until the program termination.

When we applied the GPGPU technology to the program with

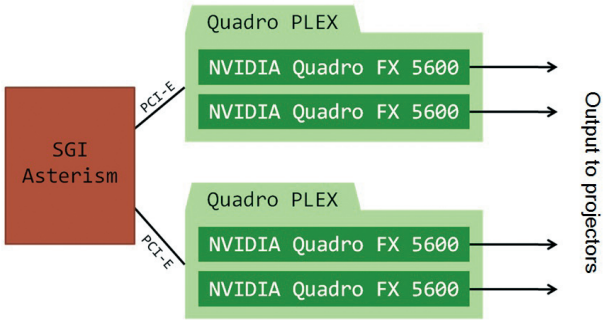

Fig. 2 The hardware configuration of graphics workstation of BRAVE.

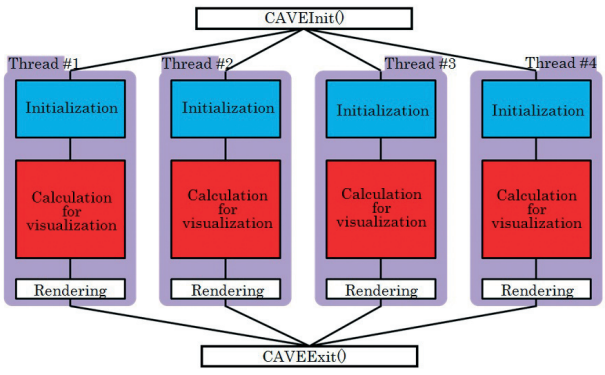

Fig. 3 The Processing flow of CAVELib program.

CAVELib, initialization process of GPU devices was added to the initialization part of each thread in Fig. 3. In the calculation part of each thread, CPU-based visualization algorithm was replaced to GPU-based one. In the case with implementation by CPU code, vertex and normal data, which were generated as execution results of visualization algorithm, are stored on main memory managed by CPU. These data must transfer to GPU memory, when the rendering process is executed. In this case, the data transfer time can become significant problem. In the case with implementation by GPU code, vertex and normal data are calculated and stored on GPU memory. These data can be used directly to render, and speeding up of the rendering process can also be expected.

Fig. 4 shows a snapshot of executing the prototype program implemented GPU-based isosurface reconstruction algorithm, and Fig. 5 shows benchmark results. As a result, it was confirmed that the processing time including drawing was greatly shortened in comparison with CPU implementation. Therefore, the effectiveness of the application of the GPGPU technology to the visualization program for the VR system was suggested. In addition to the implementation method described in this report, we are also trying to implement and evaluating the method using multi-GPU, and these methods can expect applying to other visualization software.

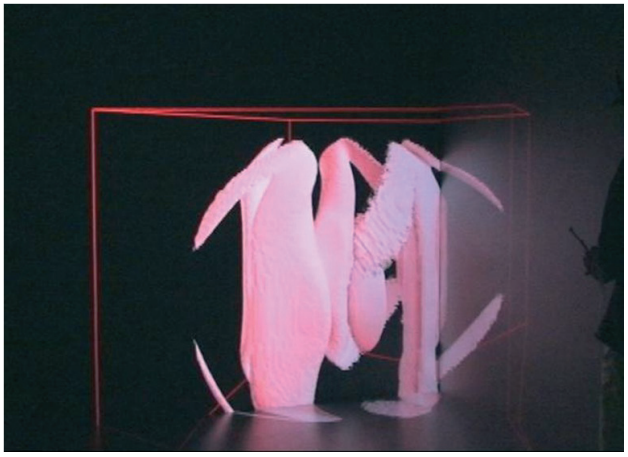

Fig. 4 A snapshot of executing the prototype program implemented GPU-based isosurface reconstruction algorithm.

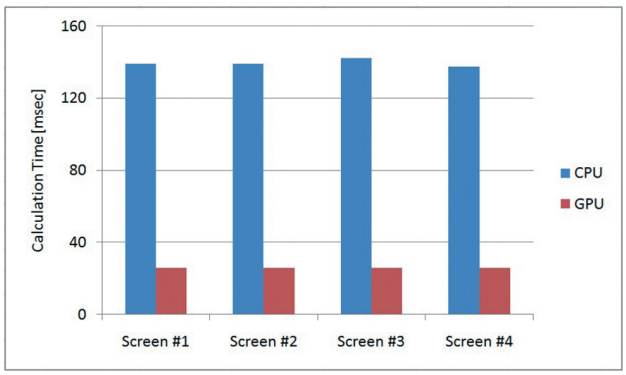

Fig. 5 Comparison of calculation time between CPU and GPU (datasize: 128x128x128).

4. Visual data mining

We developed visual data mining methods to analyze automatically large amounts of time-varying volume dataset [8]. In this study, we applied these methods to space plasma simulation data. In the field of solar terrestrial physics, it is known that the magnetic reconnection (interaction between the interplanetary magnetic field and the geomagnetic field) has a crucial role in the magnetospheric convection. The automatically classification method, which extracts magnetic field lines' topology from a highly temporal- and spatial-resolution dataset, has developed in order to understand a time-dependent change of complex magnetic field lines called magnetic flux rope. In this study, nine types of magnetic field lines are defined based on magnetic topology and the number of loop times. First, the magnetic field lines were visualized and projected onto x-y and x-z planes as shown in the lower right-hand of Fig. 6. Next, topological type of each magnetic field line is distinguished between the following three types; 'open (Sun to Earth)', 'closed (Earth to Earth)' or 'detached (Sun to Sun)', automatically by using eight scanning lines from central point located on the core field line. The number of loop times is also detected at the same time. Here, when the magnetic field line is helical, three or more intersecting points are detected on each scanning line in the x-z plane. When it is quasi-helical, one or two points are detected. When there is no flux rope, intersecting point is only one or not detected. Finally, we scan intersecting points along core field line of magnetic flux rope (drawn by dashed line as shown in Fig. 6) in the x-y plane, and classified the magnetic field line's topology as well as the x-z plane. By using 2D projected image processing, 3D time-dependent change of magnetic field line's topology of magnetic flux ropes are obtained.

By using the automatic classification method as mentioned above, a 3D distribution of magnetic field lines' topology and its separatrix surfaces are visualized. 3D distribution of magnetic field line's topology can be visualized as the 3D separatrix surfaces specified as boundary surfaces between 'open' and

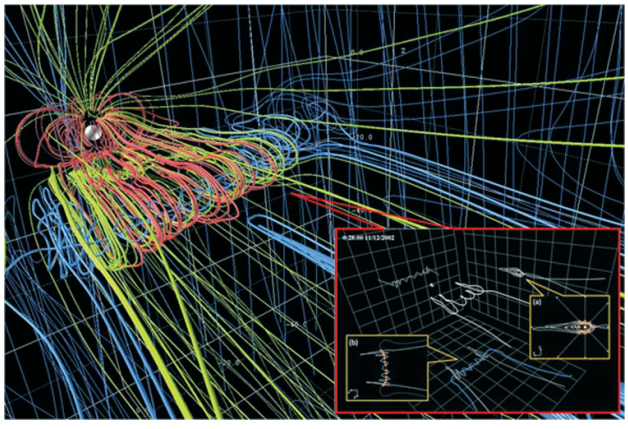

Fig. 6 Visual data mining of magnetic field line's topology.

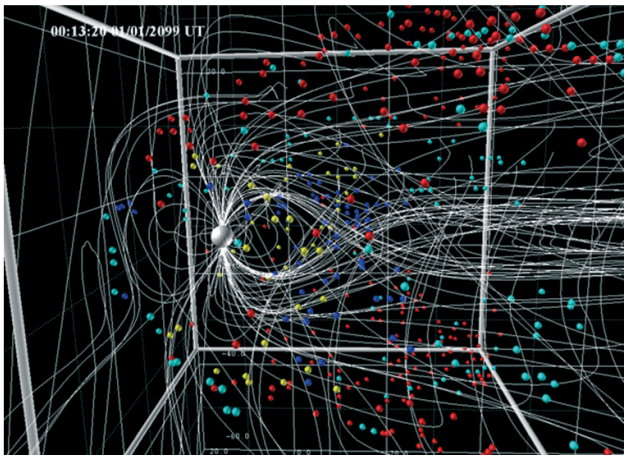

Fig. 7 Extraction of magnetic reconnection region.

'detached' field line bundles. In comparison with time-dependent change of the 3D separatrix surfaces, the magnetic reconnection region was automatically extracted as shown in Fig. 7. Red spheres represent the points at which topological type of magnetic field lines changes from 'detached' to 'open'. As the same way, blue, yellow and light-blue ones represent the points to change 'closed' to 'open', 'closed' to 'detached', and 'open' to 'detached', respectively.

5. Survey of large-scale data visualization

We investigated the studies of all sorts and sources about large-scale visualization. In this section, we introduce only the result of this survey research.

5.1 Parallel visualization

Visualization of massive datasets produced by large-scale numerical simulations takes lots of time. One solution to visualize large-scale dataset efficiently is parallelization of visualization pipeline such as filtering, mapping and rendering process. Filtering is the process to select data portion to be visualized. Data compressions (delta encoding, quantization or wavelet compression) and data structuralizing (octree, k-d tree) are known as effective filtering techniques to a handle large-scale volume dataset. Time-Space Partitioning (TSP) tree, enhanced TSP tree and wavelet TSP tree have also been developed for time-varying datasets. Volume rendering is the useful method to display a 2D projection of a 3D discretely sampled dataset. In order to handle large-scale volume dataset, many researchers have focused on its efficient algorithms. Volume rendering algorithms are roughly classified into the geometry processing and the rasterization. To achieve efficient parallel volume rendering, the sorting algorithms (sort-first, sort-middle and sort-last) during the geometry processing, the image compositing methods (direct send and binary swap) during the rasterization and other optimization techniques have been developed.

5.2 Visualization of time-varying dataset

In this case, data I/O processing occurs continuously and periodically each time step. As the efficient I/O techniques for time-varying visualization, pre-fetching, parallel I/O and parallel pipeline (temporal and spatial parallelism) are widely used on parallel computing system. In particular, parallel pipeline is widely used as effective method that can remove the I/O bottlenecks and minimize the interframe delay.

5.3 Remote visualization

The most serious bottleneck is the transferring large amounts of dataset from data storage to user's desktop machine. The method to solve this bottleneck is remote visualization that use network environment. The concepts of remote visualization are removal the bottleneck of data transferring process and utilization the available resource. Remote visualization methods are classified into several types based on server/client model and data flow model. We also investigate distributed visualization by using grid environment and collaborative visualization.

5.4 Other recent works

With the recent development of GPU, general purpose computing on GPU (GPGPU) is utilized in the visualization as well as the numerical simulation. Moreover, utilization of a GPU cluster in which each node is equipped with a GPU also reported to handle more large-scale dataset. We also investigated large-scale data visualization via GPU cluster as hybrid (tightly and loosely coupled) system and its future prospects.

6. VFIVE development

We have been developing an interactive visualization software VFIVE [2, 6, 8] for CAVE systems [3]. VFIVE has various visualization methods and its user interface is designed especially for CAVE systems unlike many of other visualization software for CAVE systems, which were designed for PCs or graphics workstations with 2D monitors. As a result of our vigorous developments, VFIVE became a powerful tool for interactive visualization.

The source code of the basic version of VFIVE has been opened on the AVPRG web site [9]. This version is executable on single systems with UNIX / Linux. Currently, VFIVE is being proceeded to tune for the PC cluster-based CAVE system with MS-Windows, which is widely installed in many laboratories of universities and research institutes in Japan.

In the fiscal year 2009, we added an animation function to handle time-developing data as shown in Fig. 8, to the original VFIVE. This extension enables users to perform 4D visualization in CAVE systems. This version is also available to download on our web site [9].

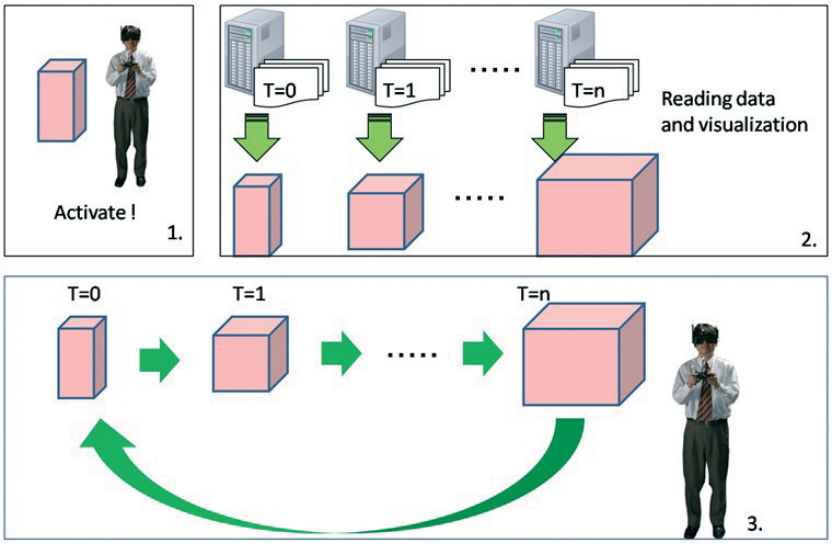

Fig. 8 VFIVE's animation function. (1) a user activate the animation function. (2) VFIVE reads data, visualizes data and saves visualized objects such as isosuface as OpenGL's display list. VFIVE does this process at all the time step. (3) VFIVE shows the saved visualized objects one after another.

7. Summery

Speeding up and vectorization of several pre-processing parts of a visualization program containing software-rendering algorithms was considered and examined. A ray-skip algorithm by octree-structured bounding volumes was equipped with pre-processing parts to accelerate ray-casting calculations. We confirmed that vectorization is effective to all of pre-processing parts and the computation time was successfully reduced by the vectorization on vector type computers.

Implementation methods of GPGPU technology to a virtual reality visualization program, especially for an isosurface reconstruction algorithm, were examined. Applications of GPGPU technology to each process of CAVELib program and memory managements were discussed. It was confirmed that the processing time including drawing was greatly shortened in comparison with CPU implementation. This result suggests that the application of GPGPU technology to the visualization program for the VR system is effective.

Visual data mining methods to analyze automatically large amounts of time-varying volume dataset of a space plasma simulation were developed and examined. These methods can classify magnetic field lines automatically into three types based on the difference between their topologies and detect the number of loop times. 3D distribution of magnetic field lines' topology and its separatrix surfaces were visualized by using this automatic classification method.

Brief overviews of the survey of recent researches and developments for large-scale data visualization were reported. The results fell roughly into four categories; parallel visualization, Visualization of time-varying dataset, Remote visualization and other recent works.

Current progress of VFIVE development was reported. The version for PC cluster-based CAVE system with MS-Windows is being tuned, which is widely installed in many laboratories of universities and research institutes in Japan. Extended version of VFIVE to handle time-developing data was opened on the AVPRG web site.

References

[1] D. Luebke, M. Harris, N. Govindaraju, A. Lefohn, M. Houston, J. Owens, M. Segal, M. Papakipos, and I. Buck, "GPGPU: general-purpose computation on graphics hardware", Proc. of 2006 ACM/IEEE Conference on Supercomputing, Tampa, Florida, USA, Nov., 2006.

[2] F. Araki, H. Uehara, N. Ohno, S. Kawahara, M. Furuichi, and A. Kageyama, "Visualizations of Large-scale Data Generated by the Earth Simulator", J. Earth Sim., Vol. 6, pp.25-34, 2006.

[3] C. Cruz-Neira, D. J. Sandin, and T. A. DeFanti, "Surroundscreen projection-based virtual reality: The design and implementation of the CAVE", Proc. of the Computer Graphics International Conference, pp.135-142, 1993.

[4] CAVELib, URL: http://www.mechdyne.com/integratedSolutions/software/products/CAVELib/CAVELCA.htm

[5] CUDA , http://www.nvidia.com/object/cuda_home.html

[6] A. Kageyama, Y. Tamura, and T. Sato, "Visualization of Vector Field by Virtual Reality", Progress Theor. Phys. Suppl., Vol.138, pp.665-673, 2000.

[7] N. Ohno and A. Kageyama, "Scientific visualization of geophysical simulation data by the CAVE VR system with volume rendering", Physics of The Earth and Planetary Interiors, Vol. 163, Issues 1-4, pp.305-311, Computational

Challenges in the Earth Sciences, 2007.

[8] D. Matsuoka, K. T. Murata, S. Fujita, T. Tanaka, K. Yamamoto, and N.Ohno, "3D Visualization and Visual Data Mining", Journal of the National Institute of Information and Communications Technology, Special Issue on Space Weather (in press).

[9] VFIVE : Virtual Reality Visualization Software for CAVE System, http://www.jamstec.go.jp/esc/research/Perseption/vfive.en.html

大規模データ可視化，GPGPU およびビジュアルデータマイニングの研究

プロジェクト責任者
荒木　文明　　海洋研究開発機構　地球シミュレータセンター
著者
荒木　文明 *1，川原慎太郎 *1，大野　暢亮 *1，松岡　大祐 *1
*1 海洋研究開発機構　地球シミュレータセンター

地球シミュレータセンター高度計算表現法研究グループで 2009 年度に実施した可視化に関する研究開発について報告する。シミュレーションと同一環境において実施する可視化手法の研究については、ソフトウェアレンダリング可視化プログラムを開発し、その中のいくつかのプリプロセスについての高速化とベクトル化を考察する。レイキャスティング計算の高速化ために、八分木で構造化されたバウンディングボリュームを用いて空領域で処理をスキップさせるアルゴリズムを提案し、可視化プログラムのプリプロセス部分への実装を行う。GPGPU 応用については、バーチャルリアリティ可視化プログラム、特に等値面再構成アルゴリズムへの GPGPU 技術の実装方法を考察し、実験および評価を実施する。このとき、描画を含む処理時間が CPU 実装の時に比べて大幅に短縮されることが確認される。他、GPU 実装に関して CAVELib に基づくプログラムの各処理との関係を議論する。データマイニングに関しては、磁力線をそのトポロジー的構造に基づいて 3 種類に分類する方法を開発する。また、この分類手法を用いて磁力線トポロジーの 3 次元分布と磁気セパラトリックス面が可視化される。近年の大規模可視化研究の調査については、その概略を報告する。また VFIVE 開発に関する現在の取り組みについても報告する。

キーワード:大規模データ可視化, ベクトル化, GPGPU, バーチャルリアリティ可視化, ビジュアルデータマイニング