G-Render: 그리드 기반 이미지 처리 시스템

김은성*, 정임영*, 최형준*, 염헌영*
*서울대학교 컴퓨터공학부
e-mail : {eskim, iyjung, hjchoi, yeom}@dcslab.snu.ac.kr

G-Render: Grid-based Image Processing System

Eunsung Kim*, Im Young Jung*, Hyung Jun Choi*, Heon-Young Yeom*
*Dept. of Computer Science & Engineering, Seoul National University

요 약

기존의 2 차원 이미지를 통한 세포 분석은 단지 세포의 단면만을 볼 수 있기 때문에 정확한 구조를 파악하기 힘들다. 본 논문에서는 그리드 기술을 이용하여 2 차원 이미지들을 세포 구조에 대한 더욱 정확한 이해 및 연구 능률의 향상을 도모할 수 있는 3 차원 이미지로 재구성하는 시스템을 개발하였다. 이 시스템은 고성능 이미지 처리를 위해서 계산 그리드를 이용하며, 화질 개선을 위한 전처리 기술, 자동 영상 정렬 기술, 효과적인 삼차원 재구성 기술과 같은 다양한 이미지 처리 알고리즘 및 preStageIn, BgUpload, delegated preprocessing 등과 같은 데이터 전송 최적화 기술 등을 제공한다. 또한, 다양한 이미지 뷰어 기능 및 DirectX 를 이용한 3 차원 렌더링 기능을 제공한다.

1. 서론

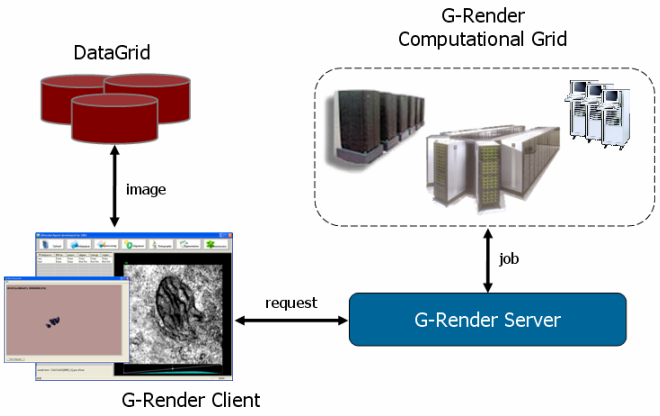
전자 현미경 영상은 세포단위 작은 입자 조직의 형태 및 구조의 관찰 및 분석을 위해 연속적으로 획득한 2 차원 이미지이다. 이러한 2 차원 이미지들은 세포의 단면만을 볼 수 있기 때문에 이 이미지들만을 이용해서는 정확한 구조를 파악하기 어렵다. 따라서 적절한 가공을 통해서 3 차원 이미지로 재구성할 수 있다면, 또한 이 과정을 매우 신속하게 수행할 수 있다면, 세포 구조에 대한 더욱 정확한 이해 및 연구 능률의 향상을 도모할 수 있어서 관련 연구 분야에 많은 도움을 줄 수 있다.

본 논문에서는 고배율 전자 현미경을 통해서 얻은 2 차원 이미지들을 3 차원 이미지로 가공해 주는 시스템인 G-Render 시스템을 구현하였다.

G-Render 시스템은 고성능 이미지 처리를 위해서 계산 그리드를 이용하며, 화질 개선을 위한 전처리 기술, 자동 영상 정렬 기술, 효과적인 삼차원 재구성 기술과 같은 다양한 이미지 처리 알고리즘 및 preStageIn, BgUpload, delegated preprocessing 등과 같은 데이터 전송 최적화 기술 등을 이용한다. 또한, 다양한 이미지 뷰어 기능 및 DirectX 를 이용한 3 차원 렌더링 기능을 제공한다.

2. G-Render 시스템 구성

G-Render 시스템은 계산 그리드 기반 이미지 프로세싱 시스템으로 전체 구성은 (그림 1)과 같다. 데이터 그리드는 G-Render 시스템에서 이용되는 이미지 파일들을 공유해서 사용할 수 있는 이미지 파일 저장소이다. G-Render 클라이언트는 G-Render 시스템에 대한 사용자 인터페이스를 제공하며, 다양한 이미지 뷰어 및 3 차원 렌더링 도구를 포함하고 있다. G-Render 서버는 G-Render 클라이언트의 이미지 처리 작업 요청을 받아서 이를 적절히 스케줄링하고 작업 프로세스를 관리하는 역할을 제공한다. 이 서버에서는 2 차원 이미지들을 이용해서 3 차원 이미지로 재구성하기 위해서 다음과 같은 3 가지 이미지 처리 방법을 제공한다.



(그림 1) G-Render 구조

- 이미지 선처리 (Image preprocessing)
  이미지의 잡음을 제거하기 위해서 선처리 작업을 수행하여 보정된 이미지를 생성한다.
- 이미지 정렬 (Image alignment)
  전자현미경의 기계적인 오차를 영상 처리 기법을 이용하여 보정하는 기술로써 클라이언트에서 전달된 영상 정렬 옵션들을 기반으로, 변형된 영상을 자동으로 보정하여 영상을 재생성 한다.
- 토모그램 생성 (Tomogram generation)

정렬된 2 차원 영상을 3 차원으로 재구성된 tomogram 을 생성

마지막으로 G-Render 시스템은 이러한 이미지 처리 방법을 신속히 수행하기 위해서 고성능의 G-Render 계산 그리드를 이용한다.

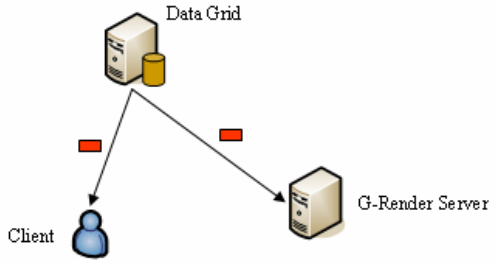
G-Render 시스템의 사용 시나리오는 다음과 같다. 사용자는 데이터 그리드를 통해서 자신이 사용할 이미지 데이터를 다운로드 한다. 이렇게 다운로드한 이미지 파일의 처리를 위해서 적당한 옵션을 입력한 후 이미지 처리 작업 요청을 G-Render 서버에 한다. G-Render 서버는 이 작업 처리 요청을 받은 후, 이를 적절히 스케줄링 한 후에 실제 작업 처리를 위해서 G-Render 계산 그리드에 작업을 분산 제출한다. 분산된 작업의 결과가 모두 취합되면 서버는 이 결과 파일에 대한 URL 을 사용자에게 리턴 한다. 사용자는 이 URL 을 통해서 G-Render 서버로부터 결과 파일을 다운로드 한다. 사용자는 다운로드한 파일을 G-Render 클라이언트의 뷰어를 통해서 살펴보고, 적당한 결과를 얻지 못했으면 새로운 작업을 G-Render 서버에 제출하게 된다. 만약 적절한 결과 이미지와 3 차원 영상을 생성했으면, 이를 다른 연구자와 공유를 위해서 데이터 그리드로 업로드할 수 있다.

3. 이미지 파일 전송 최적화

다른 그리드 시스템과 마찬가지로 G-Render 시스템은 사용자가 이미지 처리 작업을 요청할 때, 이미지 처리 작업에 사용되는 용량이 큰 파일을 G-Render 서버로 올리는 부담을 가지고 있다. 따라서 이를 줄이기 위해서 G-Render 에서는 다음과 같은 이미지 파일 전송에 대한 최적화 메커니즘을 제공한다.

(1) Pre file stageIn

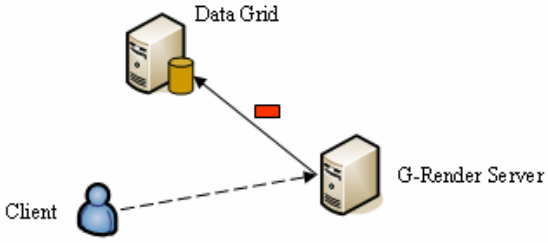
사용자가 데이터그리드를 통해서 이미지 파일을 전송받을 때, 같은 이미지 파일을 미리 G-Render 서버로 전송하여, 이미지 처리 작업 요청시 파일 stageIn 시간을 없앨 수 있다. (그림 2)는 이러한 메커니즘을 보여주고 있다.



(그림 2) PreStageIn 메커니즘

(2) Background file upload

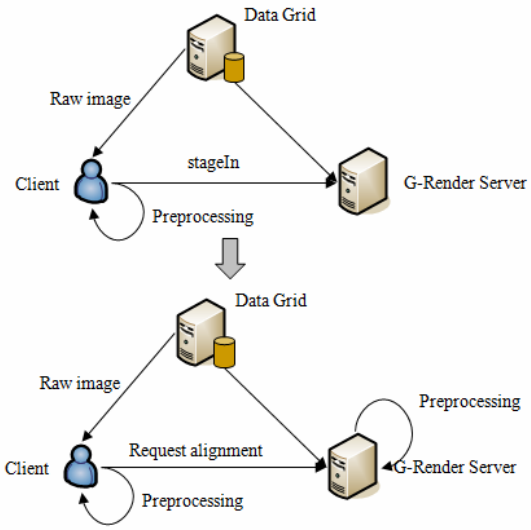
사용자가 자신이 처리한 이미지 파일 공유를 위해서 데이터 그리드로 이미지 파일을 올릴 때, 사용자 측에서 이미지 파일을 데이터 그리드로 전송하는 것이 아니라, (그림 3)에서 보는 것처럼 동일한 이미지 파일을 G-Render 서버에서 올리도록 하는 메커니즘이다.



(그림 3) BgUpload 메커니즘

(3) Delegated preprocessing

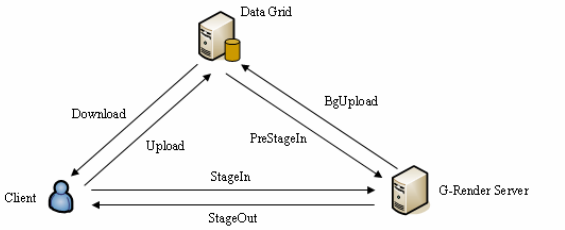
이 메커니즘은 선처리된 이미지 파일의 stageIn 을 막기 위해서 (그림 4)에서처럼 클라이언트에서 수행된 동일한 선처리 절차를 G-Render 서버에서 동일하게 처리한 후 이미지 정렬 작업을 진행하게 한다.



(그림 4) Delegated preprocessing 메커니즘

(4) 이미지 파일 전송 최적화 요약

이상과 같이 살펴본 이미지 파일 전송 최적화를 통해서 G-Render 시스템에서는 작업을 처리하기 위한 모든 이미지 파일의 stageIn 을 제거하였다. 따라서 빈번한 파일 전송 시간을 줄일 수 있어서 시스템의 성능을 향상시킬 수 있었다. (그림 5)는 G-Render 시스템이 제공하는 이미지 파일 전송 메커니즘을 보여주고, 각 메커니즘별 각 이미지 데이터의 전송 여부를 나타낸다.



	Raw	Pre	Ali	Tomo	Seg
Download	O	-	O	O	O
PreStageIn	O	-	O	X	X
Upload	-	-	X	X	O
BgUpload	-	-	O	O	X
StageIn	X	X	X	-	-
StageOut	-	-	O	O	-

(그림 5) G-Render 파일 전송

4. 결론

G-Render 시스템은 2 차원 이미지의 3 차원 렌더링을 통해서, 기존의 2 차원 이미지들만을 통한 세포단위의 작은 입자조직의 형태 및 구조의 관찰 및 분석 작업을 획기적으로 변화시킬 수 있는 필수적인 시스템이다.

이 시스템을 이용해서 사용자는 세포의 단면만을 관찰할 수 있는 것이 아니라 세포의 실제 형태 및 구조를 현실감 있게 파악할 수 있다. 또한 이러한 작업이 계산 집약적인 프로세스이어서 많은 자원 및 시간을 요구할 수 있는데, G-Render 시스템을 그리드를 기반으로 구현함으로써 이러한 제약의 많은 부분을 제거하였다.

G-Render 시스템을 통해서 바이오 및 나노와 같은, 관련 연구 분야의 질적, 양적 향상을 이룰 수 있을 것이라고 판단된다.

참고문헌

[1] I. Foster, C. Kesselman, S. Tuecke. “The Anatomy of the Grid: Enabling Scalable Virtual Organizations”, International J. Supercomputer Applications, 15(3), 2001.
[2] I. Foster, C. Kesselman, J. Nick, S. Tuecke, “The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration”, Open Grid Service Infrastructure WG, Global Grid Forum, June 22, 2002.
[3] W. Allcock, A. Chervenak, I. Foster, C. Kesselman, C. Salisbury, S. Tuecke, "The Data Grid: Towards an Architecture for the Distributed Management and Analysis of Large Scientific Datasets", Journal of Network and Computer Applications, 23:187-200, 2001.
[4] W. Allcock, J. Bester, J. Bresnahan, A. Chernevak, I. Foster, C. Kesselman, S. Meder, V. Nefedova, D. Quesnal, S. Tuecke, "Data Management and Transfer in High Performance Computational Grid Environments", Parallel Computing, 2002.
[5] W. Allcock, J. Bester, J. Bresnahan, A. Chervenak, L. Liming, S. Meder, S. Tuecke, “GridFTP Protocol Specification”, GGF GridFTP Working Group Document, September 2002.
[6] http://www.globus.org