

유체 애니메이션 기술의 현황 및 전망

송오영 (세종대학교)

차례

1. 서론
2. 유체 애니메이션 기법 동향
3. 유체 애니메이션 기법 전망
4. 결론

1. 서론

1.1 유체 애니메이션 기술 개발의 필요성

최근 많은 영화와 게임에서 물, 불, 연기, 폭발 등의 유체 표현은 실제 현상과 거의 같도록 연출함으로써 사실감을 극대화시키고 있다. 이러한 유체 특수 효과 장면을 연출할 수 있게 된 것은 컴퓨터 그래픽스 기술 중 하나인 유체 애니메이션 기술이 발전한 덕택이다. 유체 애니메이션 기술은 키프레임이나 모션 캡처 방식 등의 일반적인 애니메이션 기법과는 다른 방식으로 발전해 왔다. 왜냐하면 유체 현상의 특성상 전통적인 기법으로 해결할 수 없는 어려움이 있기 때문이다. 먼저, 유체 현상은 정해진 형태가 없이 매우 다양하게 변화하므로 수작업으로 움직임은 **사실적으로 만들어 내는 것은 거의 불가능하다**. 결국, 유체 현상을 재현하기 위해서는 물리 기반 시뮬레이션 기법을 도입하여야 하는데, 이는 **매우 많은 계산 시간을 요구한다**. 또한, 여타의 물리 기반 시뮬레이션보다 높은 자유도를 갖는 유체의 특성상 **애니메이션의 제어가 훨씬 더 어렵다**는 단점이 있다. 유체 애니메이션 기술은 이러한 세 가지 어려움을 극복하는 방향으로 발전되어 왔다.

1.2 유체 애니메이션 기술의 목표

일반적으로 많은 사람들은 유체 애니메이션의 목표를 실제 현상과 구별할 수 없을 정도로 똑같이 시뮬레이션 하는 것으로 알고 있다. 하지만 이는 주로 유체역학, 음용 수학 관점에서 추구하는 바이고, 컴퓨터 그래픽스 관점에서 본다면 물리적 사실성외에도 앞에서 언급했던 어려움을 극복해야 하는 과제도 안고 있다. 실제로 컴퓨터 그래픽스 기술로서의 유체 애니메이션은 다음과 같은 세 가지

목표를 추구한다.

(1) **시각적 사실성** : 결과가 시각적으로 사실적이어야 한다. 이는 물리적 사실성과는 다소 구별되는 개념으로 물리식에 기반을 둔 방법이 아니라도 시각적으로 사실적인 결과를 만들어 낼 수 있는 방법도 허용할 수 있는 관점이다. 물론 자연현상을 물리 방정식으로부터 시뮬레이션 하는 것만으로 그대로 재현하는 것이 가장 좋겠지만, 눈으로 보기에 사실적인 결과를 만들어내는 것으로 목표를 정한다면 등원할 수 있는 방법에 있어 더 많은 옵션을 가질 수 있게 된다. 정확한 물리 방정식이 알려지지 않은 현상이나 매우 복잡한 현상(거품, 점탄성 유체 등)을 효율적으로 재현하는 방법을 모색할 수 있다.

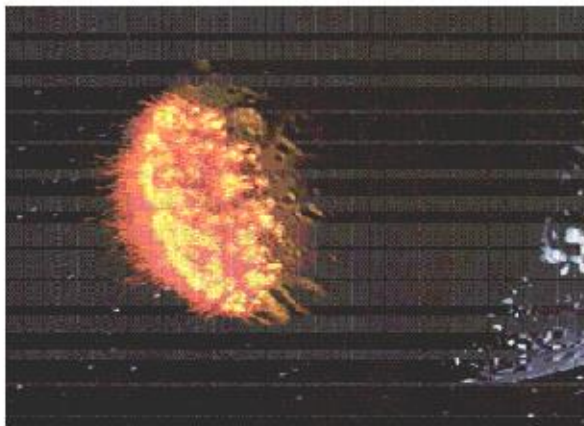
(2) **시뮬레이션 안정성(신속성)** : 큰 시뮬레이션 시간 간격(Δt), 공간 간격(Δx)을 사용할 수 있어야 한다. 안정성은 다른 공학 분야에서도 중요한 덕목이지만, 결과의 정확성을 기하기 위해 큰 Δt , Δx 의 사용은 일부러 자제하는 상황이다. 유체 애니메이션에서는 계산 속도와 사용되는 메모리의 양을 줄이기 위해 일반적으로 다른 공학 분야에 비해 훨씬 큰 Δt , Δx 를 사용하게 된다. 그러므로 안정성이 대단히 중요한 문제로 떠오르게 된다.

(3) **제어가능성** : 사용자가 원하는 대로 결과를 만들 수 있어야 한다. 영화 「반지의 제왕」에서 담배 연기를 뽑아내었을 때 연기가 배의 형태를 띠는 장면이 소개되었다. 이러한 기능은 사실성과 상충되는 면이 있으나 사실성을 최대한 훼손하지 않도록 이루어져야하며 영화·애니메이션, 게임 제작에서 높은 활용 가치를 가질 수 있다.

다음 장은 제시된 목표를 유체 애니메이션 기술이 어떠한 방식으로 달성해 왔는지 살펴본다.

2. 유체 애니메이션 기법 동향

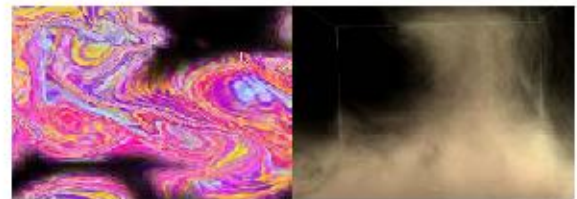
유체 애니메이션은 1983년 Reeves 논문[1]에서 입자 시스템 기반으로 행성 충돌 장면을 표현한 이후로 많은 발전이 있었다(그림 1) 이후 입자 기반의 절차적 방법으로 유체를 표현하다가 1990년대 후반부터 전산유체역학(CFD: Computational Fluid Dynamics)분야의 수치 기법들이 도입되기 시작하였다. 이미 오래전부터 전산유체역학에서는 유체의 움직임은 지배하는 물리 방정식인 나비에-스톡스 방정식(Navier-Stokes Equation)의 해를 구하는 다양한 수치적 해법들을 개발해왔다. 하지만 전산유체역학의 기법들은 대체로 많은 계산량을 요구하며 시뮬레이션 시간간격이 큰 경우 안정성을 보장하지 못한다.



▶▶ 그림 1. 1983년 Reeves 논문의 행성 충돌 장면

1999년 Stam의 ‘Stable Fluids’ 논문[2]이 발표됨으로써 본격적으로 유체 애니메이션의 안정성을 보장할 수 있게 되었다. 이 논문은 시뮬레이션 시간 간격이나 공간 격자의 크기에 관계없이 항상 안정적으로 유체를 시뮬레이션 하는 방법을 소개하였다. 큰 시간 간격과 큰 격자 크기로 인해 물리적인 정확성은 희생하더라도 유체 특유의 시각적인 사실성은 보장하고 사용자의 상호작용을 중요시한 측면이 여타의 전산유체역학에서의 기법과 다른 점이다(그림 2) 이 논문을 시작으로 전산유체역학의 목표와는 다른 컴퓨터 그래픽스에서의 유체 애니메이션의 목표가 확실히 제시되었다. 2001년 Fedkiw등은 ‘Stable

Fluids’ 논문이 제시한 안정성은 보존하면서 시각적 사실성을 향상시킨 논문을 발표하였다[3]. 이 논문에서는 고차의 보간법을 사용하여 나비에-스톡스 방정식의 대류항의 해를 구하는 방법을 제시하였으며, 와류 고정법(vorticity confinement)을 도입하여, 성긴 격자에서도 기체 현상의 특징인 소용돌이를 효과적으로 재현하였다(그림 3). 와류 고정법은 물리 방정식으로부터 기인한 방법이라기보다는 시각적으로 그럴듯한 결과를 보여주기 위해 고안된 방법이다. 이러한 접근방법은 이후의 유체 애니메이션 기술의 방법론에 적지 않은 영향을 끼쳤다. 이 두 논문은 기체 현상을 다루는 연구의 기반이 되었다.



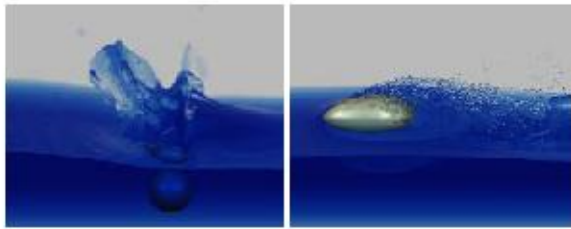
▶▶ 그림 2. 1999년 Stam의 ‘Stable Fluids’ 연구 결과



▶▶ 그림 3. 2001년 Fedkiw등의 연기 애니메이션 결과

2000년대에 들어와서는 액체 현상 또한 물리적인 기법은 통해 애니메이션하려는 시도가 시작되었다. 액체 현상은 나비에-스톡스 방정식의 수치적 해법 외에 물표면의 움직임을 추적하는 방법이 필요하다. 2001년에 Foster 등이 물 표면을 추적하는 방법으로 레벨 셋 방법[4]에 입자를 혼합한 기법을 제안하였다[5]. 레벨 셋 방법은 암시적 표면 표현법(implicit surface representation)으로 액체 표면으로부터 부호-거리(signed distance)를 격자점에 저장하는 방식이다. 매 시뮬레이션 시간마다 격자점의 레벨 셋이 0이 되는 값을 구하면서 액체 표면을 추적한다. 레벨 셋 방법만으로는 격자가 성길수록 액체 표면을 정확히 추적하기 어려워지기 때문에 Foster 등의 연구에서는 표면 가까운 곳에 액체 입자를 생성하여 표면을 보다 정확

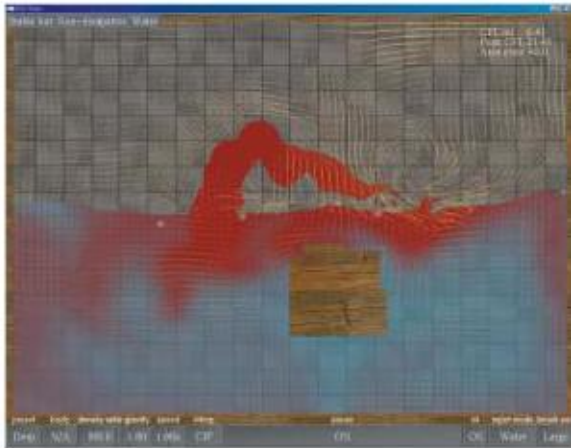
히 추적하였다. 그 결과, 잔잔하게 움직이는 물 표면은 비교적 정확하게 추적할 수 있었으며, 물방울 튀김 현상도 쉽게 생성할 수 있었다(그림 4). 2002년에 Enright 등이 액체 표면 안쪽뿐만 아니라 바깥쪽에도 입자를 설정함으로써 훨씬 정교한 표면 추적을 할 수 있는 방법(입자 레벨 셋: Particle Level Set)을 제안하였다[6](그림 5). 입자 레벨 셋은 후속 유체 연구에서 표면을 추적하는 기법의 표준이 되었다.



▶▶ 그림 4. 2001년 Foster 등의 액체 현상 및 물방울 효과



▶▶ 그림 5. 2002년 Enright 등의 액체 애니메이션 연구 결과

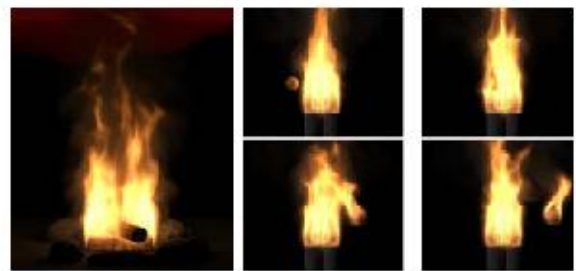


▶▶ 그림 6. 2005년 Song 등의 연구 결과: 2차원 실시간 다상 유체 시뮬레이션

하지만 액체 현상은 다른 논문들은 기체현상과는 달리, 계산 시간이 많이 걸렸다. 이는 정확한 액체 표면의 위치를 추적하기 위해 보다 작은 시간 간격을 써야 하거나, 고

차의 수치적 해법을 사용해야 하기 때문이다. 이를 극복하기 위해 안정성을 보장하여 계산 시간을 상당히 줄인 논문은 Song 등이 2005년에 발표하였다[7]. 이 연구에서는 실시간에 상호작용적으로 동작하는 2차원 다상 유체 현상을 시뮬레이션 하였으며, 3차원의 경우에도 이전의 액체 논문보다 약 20배 빠른 시뮬레이션 성능을 보였다(그림 6).

기체 및 액체 현상을 위한 안정적인 시뮬레이션 방법이 확립됨에 따라, 이를 바탕으로 유체의 다양한 현상을 표현하려는 시도가 생겨났다. 2002년 Nguyen 등은 불꽃 면을 추적하기 위해 레벨 셋을 이용하였다[8]. 연료가 가스 혼합물로 변하는 경계면을 레벨 셋으로 추적하는 방법으로 사실적인 불 모양을 생성해 냈다(그림 7). 2007년 Hong 등은 DSD (Detonation Shock Dynamics) 기법을 레벨 셋 방식식에 적용하여 격렬한 불꽃을 생성하였다[9](그림 8). 2003년 Feldman 등은 폭발 현상을 비압축 유체 시뮬레이터에 입자기법을 도입하여 효율적으로 애니메이션 하였다[10](그림 9). 폭발은 가장 격렬한 연소 현상으로 비압축 유체 현상이라기보다는 압축 유체 현상이다. 이를 정확히 풀려면 훨씬 많은 계산 시간이 요구되므로 Feldman 등은 물리적 엄밀성을 희생하면서도 비압축 유체의 지배방정식을 약간 변형하는 방법으로 접근하였다.



▶▶ 그림 7. 2002년 Nguyen 등의 불 현상 연구 결과

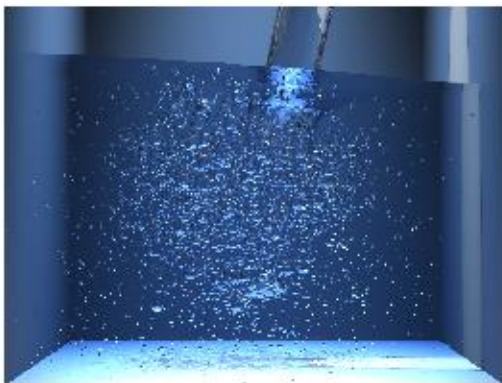


▶▶ 그림 8. 2007년 Hong 등의 불 현상 연구 결과



▶▶ 그림 9. 2003년 Feldman 등의 폭발 현상 연구 결과

기체와 액체가 혼합되면서 생기는 현상으로 빠질 수 없는 것이 공기방울과 거품이다. 이 현상은 격자 기반 방법만으로 추적하기 어려운 현상으로 입자기법을 적극적으로 도입한 방법이 소개되었다. 2008년 Hong 등은 격자 기반의 유체 시뮬레이터에 SPH (Smoothed Particle Hydrodynamics) 방법을 도입하였다[11]. Hong 등은 공기방울이 소용돌이치면서 올라오는 움직임을 재현하기 위해, 새로운 좌류 고정법을 소개하였다(그림 10). 2007년 Cleary 등은 SPH로 매우 사실적인 맥주 거품 현상을 재현하는 것에 성공하였다[12](그림 11).



▶▶ 그림 10. 2008년 Hong 등의 공기 방울 애니메이션 연구 결과

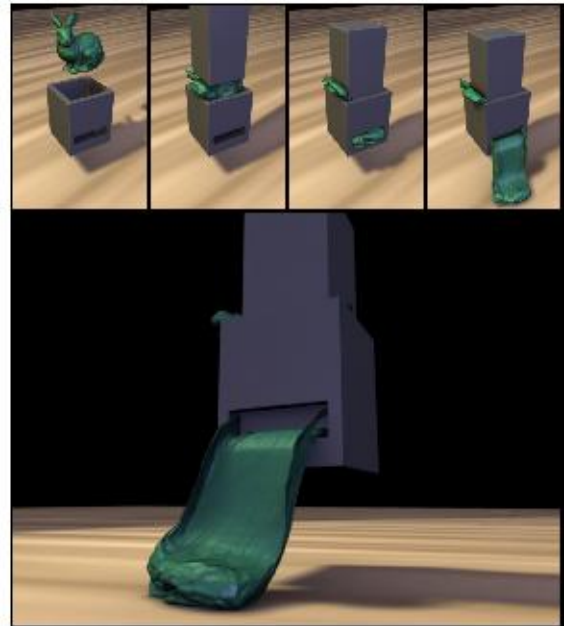


▶▶ 그림 11. 2007년 Cleary 등의 거품 애니메이션 연구 결과

우리가 일상적으로 여기는 유체와는 달리 점성뿐만 아니라 탄성을 갖는 유체를 시뮬레이션 하려는 시도도 계속되었다[13][14](그림 12, 그림 13). 점탄성 유체는 나비에-스톡스 방정식에 탄성을 나타내는 항을 더하여 수치적인 해를 구하는 방법으로 시뮬레이션 한다.



▶▶ 그림 12. 2004년 Goktekin 등의 점탄성 유체 연구 결과



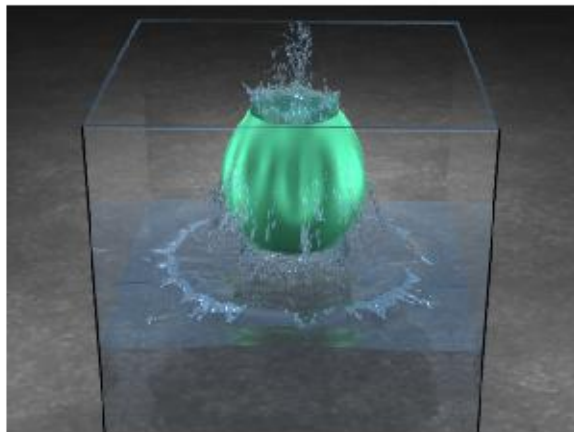
▶▶ 그림 13. 2008년 Wojtan의 점탄성 유체 애니메이션 결과

실제 영화나 게임에서 보면 유체 현상 단독으로 존재하는 경우는 드물다. 따라서 유체 현상만이 아니라 다른 객체와의 상호작용도 고려되어야 한다. 2004년 Carlson 등은 고체와의 상호작용을 다룬 논문을 발표하였고[15](그림 14). 2005년과 2008년 스텐포드대학의 Fedkiw 그룹은 고체뿐만 아니라 옷감이나 풍선과 같은 변형체와의 상호작

음을 다룬 논문을 발표하였다[16][17](그림 15). 2008년 Lenaerts는 스펀지와 같이 액체를 잘 흡수하는 물체와 액체 사이의 상호작용을 다룬 논문을 발표하였다[18](그림 16).



▶▶ 그림 14. 2004년 Carlson 등의 고체와 액체의 상호작용 결과



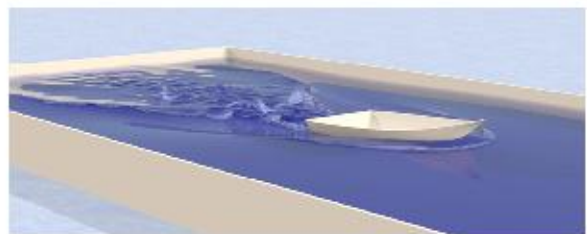
▶▶ 그림 15. 2008년 스탠포드 대학의 Fedkiw 그룹의 변형체와 액체의 상호작용 결과



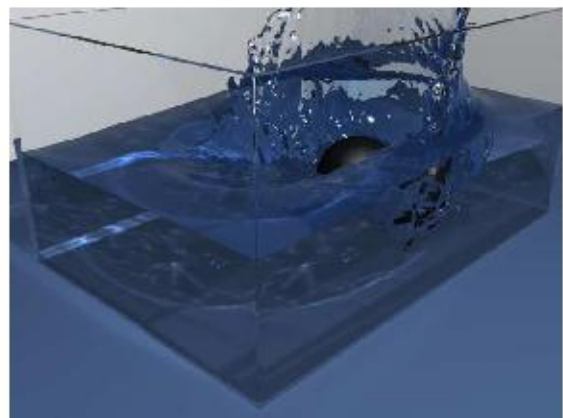
▶▶ 그림 16. 2008년 Lenaerts 등의 흡수성 물체와 액체의 상호작용 결과

앞의 방법들로 개발된 시뮬레이터가 다양한 현상을 표현 할 수 있을지라도 거대 규모의 유체 현상을 표현 하려

면 많은 계산 시간을 요구하게 된다. 이를 효율적으로 시뮬레이션 하기 위한 방법도 소개되었다. 2004년 Losasso 등은 균일한 격자가 아닌 옥트리(octree) 구조의 격자에서 시뮬레이션을 수행하는 기법을 제시하였다[19]. 이는 수면이나 물체와의 경계 근처에서는 정확히 풀고 그렇지 않은 부분의 정확도를 희생하는 방법으로 효율성을 증대 시켰다. 2006년 Irving 등은 시뮬레이션에서의 보트 움직임과 같이 수면에서 주로 일어나는 현상을 효율적으로 재현하기 위해 수면에서만 3차원 격자를 유지하고, 그 외의 공간에서는 2차원 높이-장(Height-Field) 방식의 격자를 쓰는 기법을 제시하였다[20](그림 17). 2007년 Song 등은 속도가 매우 빠르고 복잡한 유체 현상을 재현하기 위해, 옥트리 기반의 시뮬레이터에 미분 입자(Derivative Particle)를 도입하는 방법을 제시하였다[21](그림 18). 미분 입자는 물리량은 지닌 입자로 실제 값과 더불어 공간적 미분치를 갖는다. 따라서 입자 주변의 세밀한 현상을 더 정확히 모사할 수 있다. 2008년 Kim 등은 기존의 유체 시뮬레이터에 wavelet 방식의 난류 모델을 적용하여, 격자가 많지 않음에도 복잡한 난류현상을 애니메이션 하였다[22](그림 19).



▶▶ 그림 17. 2006년 Irving 등의 2차원 기법과 3차원 기법을 혼합하여 만든 결과

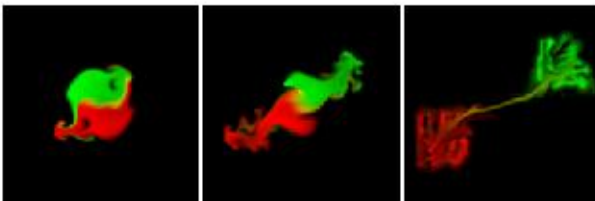


▶▶ 그림 18. 2007년 Song 등의 빠른 속도의 물 움직임 시뮬레이션 결과



▶▶ 그림 19. 2008년 Kim 등의 wavelet 방식의 난류 모델 결과

지금까지 살펴본 연구 결과는 주로 유체의 안정성 및 사실성 향상을 목표로 하였다. 한편, 유체 애니메이션의 또 다른 중요한 목표인 제어가능성을 향상시키려는 연구도 꾸준히 진행되어왔다. 2004년 Patal 등은 나비에 스톡스 방정식의 외력을 조절하여 원하는 모양의 연기가 생성되도록 하는 방법을 제안하였다[23]. 이 연구는 나비에-스톡스 방정식을 그대로 풀기 때문에 사실성을 크게 훼손하지 않으면서도 연기를 제어할 수 있었다(그림 20). 2003년과 2004년에 워싱턴 대학의 Popovic 그룹은 최적화 기법을 이용하여 키프레임 방식으로 유체의 움직임은 제어하였다[24][25](그림 21). 하지만 이 방법은 최적화 기법을 사용하기 때문에 계산속도가 느리다는 단점이 있다. 2008년 Dobashi 등은 구름의 모양을 제어하여 원하는 모양으로 구름이 생성될 수 있는 방법을 제안하였다[26](그림 22).



▶▶ 그림 20. 2004년 Fattal 등의 연기 컨트롤 애니메이션 결과



▶▶ 그림 21. 2004년 워싱턴 대학 Popovic 그룹의 유체 컨트롤 결과



▶▶ 그림 22. 2008년 Dobashi 등의 구름 생성 시뮬레이션 결과

3. 유체 애니메이션 기법 전망

현재 유체 애니메이션 연구는 국외뿐만 아니라 국내에서도 활발히 연구되고 있다. 이 논문에서는 주로 격자 기반으로 푸는 유체 애니메이션 기술의 현황을 알아보았으나 앞으로는 격자와 입자를 적절히 혼합하여 사용하는 방법들이 많이 각광 받게 될 것으로 보인다. 또한 하드웨어가 병렬화 됨에 따라 유체 시뮬레이션 알고리즘도 병렬화가 가속되고 있다. 이로 인해 계산 속도의 획기적인 향상이 이루어질 것이며, 조만간 대규모의 폭포, 홍수, 폭발 장면을 실시간에 시뮬레이션 할 수 있을 것이다. 마지막으로 유체 애니메이션의 제어 가능성에 관한 연구는 다른 연구에 비해 성과가 적은 편인데, 앞으로는 보다 편리하게 유체를 제어하는 연구가 활발히 이루어질 것으로 보인다. 오래지 않아 사용자가 자유자재로 유체의 움직임을 제어할 수 있는 기술이 개발되어 우주의 왕관현상과 같이 작은 규모의 아름다운 유체 모습을 그럴듯하게 생성해 낼 수 있을 것으로 보인다.

4. 결론

본 논문은 최근의 유체 애니메이션 기술의 현황을 살펴본 뒤, 앞으로의 연구방향에 대해서 전망해 보았다. 유체 애니메이션 기술은 영화나 게임의 특수효과에 더욱더 활발히 사용되어 표현의 범위를 넓힐 것이다. 더 나아가 영상 산업뿐만 아니라 해양, 지구환경 산업 등의 유체 응용 분야로의 파급효과가 있을 것으로 기대한다.

참고문헌

- [1] W.T. Reeves, "Particle System -A Technique for Modeling a Class of Fuzzy Objects," in Proc. of SIGGRAPH '83, July 1983, pp.359-376.
- [2] J. Stam, "Stable Fluids," in Proc. of SIGGRAPH '99, 1999, pp.121-128.
- [3] R. Fedkiw, J. Stam, and H.W. Jensen, "Visual Simulation

- of Smoke," in Proc. of SIGGRAPH 2001, 2001, pp.15-22.
- [4] Osher, S. and Fedkiw, R. The Level Set Method and Dynamic Implicit Surfaces. Springer-Verlag, New York, 2002.
- [5] N. Foster and R. Fedkiw, "Practical Animation of Liquids," in Proc. of SIGGRAPH 2001, 2001, pp.23-30.
- [6] D. Enright, S. Marschner, and R. Fedkiw, "Animation and Rendering of Complex Water Surfaces," in Proc. of SIGGRAPH 2002, July 2002, pp.736-744.
- [7] Oh-young Song, Hyuncheol Shin, and Hyeong-Seok Ko, "Stable but Nondissipative Water," ACM Transactions on Graphics 2005, Vol. 24, No.1, pp 81-97.
- [8] D. Nguyen, R. Fedkiw, and H.W. Jensen, "Physically Based Modeling and Animation of Fire," in Proc. of SIGGRAPH 2002, 2002, pp.721-728.
- [9] Jeong-Mb Hong, Tamar Shinar and Ron Fedkiw, "Wrinkled Flames and Cellular Patterns," ACM Transactions on Graphics (In Proceedings of ACM SIGGRAPH 2007), Volume 26, Issue 3, pp.471-476, 2007.
- [10] B. E. Feldman, J. F. O'Brien, and O. Arikan, "Animating Suspended Particle Explosions," in Proceedings of SIGGRAPH 2003, Annual Conference Series, (San Diego, California), pp. 708-715, July 2003.
- [11] Jeong-Mb Hong, Ho-Young Lee, Jong-Chul Yoon and Chang-Hun Kim, "Bubbles Alive," ACM Transactions on Graphics (In Proceedings of ACM SIGGRAPH 2008)
- [12] Paul W. Cleary, Soon Hyoung Pyo, Mahesh Prakash and Bon Ki Koo, "Bubbling and Frothing Liquids," ACM Trans. on Graphics 26(3), SIGGRAPH 2007.
- [13] T. Goktekin, A. W. Bargteil, and J. F. O'Brien, "A Method for Animating Viscoelastic Fluids," in Proceedings of SIGGRAPH 2004, Annual Conference Series, (Los Angeles, California), pp. 463-468, Aug. 2004.
- [14] Chris Wojtan and Greg Turk, "Fast Viscoelastic Behavior with Thin Features," ACM Transactions on Graphics (In Proceedings of ACM SIGGRAPH 2008)
- [15] Mark Carlson, Peter J. Micha and Greg Turk, "Rigid Fluid: Animating the Interplay Between Rigid Bodies and Fluid," ACM Transactions on Graphics (In Proceedings of ACM SIGGRAPH 2004)
- [16] Guendelman, E., Selle, A., Losasso, F., and Fedkiw, R. 2005. "Coupling water and smoke to thin deformable and rigid shells," ACM Trans. Graph. (SIGGRAPH Proc.) 24, 3, 973 - 981.
- [17] Avi Robinson-Mosher, Tamar Shinar, Jon Gretarsson, Jonathan Su, and Ron Fedkiw, "Two-way Coupling of Fluids to Rigid and Deformable Solids and Shells," ACM Transactions on Graphics (In Proceedings of ACM SIGGRAPH 2008)
- [18] Toon Lenaerts, Bart Adams, and Phil Dutré, "Porous Flow in Particle-Based Fluid Simulations," ACM Transactions on Graphics (In Proceedings of ACM SIGGRAPH 2008)
- [19] Frank Losasso, Frederic Gibou, and Ron Fedkiw, "Simulating Water and Smoke with an Octree Data Structure," in Proc. of SIGGRAPH 2004, 2004.
- [20] Irving, G. Guendelman, E., Losasso, F., and Fedkiw, R., "Efficient Simulation of Large Bodies of Water by Coupling Two and Three Dimensional Techniques," in Proc. of SIGGRAPH 2006, 2006.
- [21] Oh-young Song, Doyub Kim, and Hyeong-Seok Ko, "Derivative Particles for Simulating Detailed Movements of Fluids," IEEE Transactions on Visualization and Computer Graphics 2007, Vol. 13, No.4, pp 711-719.
- [22] Theodore Kim, Nils Thürey, Doug James, and Markus Gross, "Wavelet Turbulence for Fluid Simulation," ACM Transactions on Graphics (In Proceedings of ACM SIGGRAPH 2008)
- [23] Raanan Fattal and Dani Lischinski, "Target Driven Smoke Animation," in Proc. of SIGGRAPH 2004, 2004.
- [24] A. Treuille, A. McNamara, Z. Popovic, and J. Stam, "Keyframe Control of Smoke Simulations," in Proc. of SIGGRAPH 2003, July 2003, pp.716-723.
- [25] Antoine McNamara, Adrien Treuille, Zoran Popovic, and Jos Stam, "Fluid Control Using the Adjoint Method," in Proc. of SIGGRAPH 2004, 2004
- [26] Yoshinori Dobashi, Katsutoshi Kusumoto, Tomoyuki Nishita, and Tsuyoshi Yamamoto, "Feedback Control of Cumuliform Cloud Formation based on Computational Fluid Dynamics," ACM Transactions on Graphics (In Proceedings of ACM SIGGRAPH 2008)

저자 소개

● 송 오 영(Oh-young Song)



- 1998년 2월 : 서울대학교 전기공학부 (공학사)
- 2000년 2월 : 서울대학교 전기컴퓨터공학과 (공학석사)
- 2004년 8월 : 서울대학교 전기컴퓨터공학과 (공학박사)

• 2008년 4월 ~ 현재 : 세종대학교 디지털콘텐츠학과 조교수

<관심분야> 컴퓨터 그래픽스, 물리 기반 애니메이션, 전산 유체 역학