

카툰 스타일 게임을 위한 환경 적응적 실시간 유체 렌더링

김광태*, 박경주**

중앙대학교 첨단영상대학원

kgthtj@gmail.com, kjpark@cau.ac.kr

Environment-Adaptive and Real-Time Rendering of Fluids for Cartoon Style Game

Kwang-tae Kim*, Kyoung-ju Park**

Graduate School of Advanced Imaging Science Multimedia and
Film Chung-Ang University

요 약

본 논문에서는 장면에 어울리도록 유체를 카툰 스타일로 표현할 수 있는 실시간방법을 제안하고자 한다. 실제로 유체는 석양, 낮, 밤에 서로 다른 색상 톤을 보여주고 있고 수심이 낮은 곳에서의 반사 굴절 정도와 수심이 깊은 곳에서의 정도가 상이하며 색상 조차 다르다. 본 논문에서는 광원 수심 지형 시점 등에 적응적으로 유체를 카툰 스타일로 표현하는 방식을 제안한다. 실험은 기존의 방법을 구현하고 제안된 결과와 비교 분석하였다. 이러한 연구는 기존의 카툰 스타일렌더링 방법과 제안하는 카툰 스타일 방법과 동시에 사용하면 게임을 카툰 스타일로 표현할 수 있는 전체 시스템을 제시할 수 있다.

ABSTRACT

This paper to the scene to fit the fluid can be expressed as cartoon-style real-time method is proposed. As a matter of fact the fluid color in the simulation in which both reflection refraction and color tones are different in both cases. This paper light, water, terrain, time, etc. with cartoon-style representation of the fluid, adaptive way is proposed. Experimental implementation of the existing and proposed methods were compared with the results. This cartoon-style rendering of the existing research methods and the proposed method and at the same time using cartoon-style games that can be represented by the entire system could offer.

Keywords : Cartoon Rendering, Liquid, Base Texture, Reflection, Refraction

Received: Jul. 27, 2012 Accepted: Agu. 20, 2012
Corresponding Author: Kyoung-ju Park(Chung-Ang University)
E-mail: kjpark@cau.ac.kr

© The Korea Game Society. All rights reserved. This is an open-access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>), which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

ISSN: 1598-4540

1. 서론

카툰 렌더링 또는 셀(cel)렌더링은 카툰 스타일을 표현하기 위하여 3차원 물체의 음영을 단순화시키는 렌더링 기법으로서 비사실적 렌더링(NPR : Non-Photorealistic Rendering)의 대표적인 연구 분야이다[1]. 이러한 스타일이 유명해진 한 가지 이유는 McCloud가 그의 저서인 “만화의 이해 (Understanding Comics)[2]”에서 단순화를 통한 강조라 지칭했기 때문이다. 복잡한 것들을 없애고 단순화시킴으로써 표현하고자 하는 정보의 효과를 극대화 할 수 있다[3]. 카툰 렌더링의 원리는 램버티안(Lambertian) 셰이딩 모델을 확장한 것으로서 조명을 받는 부분과 그림자진 부분을 1차원 텍스처(texture)를 사용하여 음영을 단순화 시킨다.

유체를 카툰 스타일로 렌더링 하는 연구는 Eden et al.[4]과 You et al.[5]등 근래에도 다양하게 연구되어져 왔는데, 원래의 카툰 렌더링 방법과 달리 유체의 특징을 고려하였다. Eden et al.[4]은 처음으로 유체 카툰 렌더링 방법을 제안하였고 기존의 카툰 렌더링 기법에서 액체의 투명성을 이용하여 유체를 카툰 스타일로 표현했다. You et al.[5]는 Eden et al.[4]의 방식에 유체의 반사 굴절의 특징을 고려하여 유체의 카툰 스타일 렌더링 기법을 제안했다.

카툰 렌더링 방식을 게임에 적용시키면, 유저가 플레이 할 때 감각적이고 재미있는 느낌을 체감할 수 있다. 이러한 방식이 적용된 대표적 게임은 썰 온라인[6]과 마비노기 온라인[7]이 있다. 두 게임 모두 카툰 렌더링으로 표현하기 위해 단순화된 텍스처와 동일한 굵기의 외곽선을 사용하였지만, 썰 온라인은 두 가지 톤의 음영만을 사용하였고, 마비노기 온라인의 음영은 경계부분이 부드럽게 표현되었다.

게임에서 유체 렌더링은 아르고 온라인[8]과 월드 오브 워크래프트[9]등 게임에서 구현한 사례처럼, 투명상태의 반사와 굴절을 사용한다. 온라인 게임의 표현 스타일에서 고체는 색상 단순화, 모양

단순화 등 카툰 렌더링 기법을 적용하였지만 강, 호수 같은 유체에는 카툰 렌더링 기법을 적용하지 않았다.

본 논문에서는 장면 설정에 어울리도록 유체를 실시간 카툰 스타일로 표현할 수 있는 실시간 방법을 제안하고자 한다. 유체는 석양, 낮, 밤에 서로 다른 색상 톤을 보여주고 있고, 수심이 낮은 곳에서의 반사 굴절 정도와 수심이 낮은 곳에서의 반사 굴절 정도가 상이하며 색상조차 다르다. 따라서 광원, 수심, 지형, 시점 등에 적응적으로 유체를 카툰 스타일로 표현하는 방식을 제안한다. 유체의 기본 색상은 광원의 밝기와 위치 수심에 따라서 메쉬의 색상이 결정된다. 또한 반사 굴절 정도를 수심에 따라 변화하여 얇은 물과 깊은 물에서의 반사 굴절을 표현한다.

본 논문에서 제시하는 카툰 스타일 방법과 기존의 고체 카툰 스타일 렌더링 연구를 동시에 사용하면 게임을 카툰 스타일로 표현할 수 있는 전체 시스템을 제시할 수 있다.

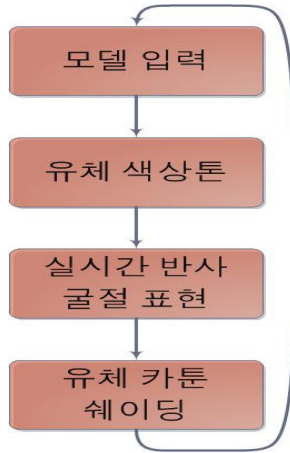
본 논문은 다음과 같이 구성된다. 2장 본문에서는 제안된 방법의 렌더링 파이프라인을 설명하며, 유체 색상 톤과 실시간 반사 굴절 표현, 유체 카툰 셰이딩 모델 생성 방법에 대해 설명한다. 3장 실험 환경 및 결과에서 결과 생성을 위한 메쉬 생성 방법과 본 논문에서 제안하는 방법의 실험 결과와 기존에 제안된 방법과 비교 분석한다. 마지막으로 결론 및 향후 연구 방향에 대해 논한다.

2. 본론

2.1 시스템 구성도

제안하는 시스템은 [Fig. 1]과 같이 테스트를 위해서 모델 입력을 받고, 환경인 수심에 따른 원색과 빛의 입사량(광원의 위치 밝기 개수, 산란 고려)에 따른 원색을 이용하여 유체 색상을 페이스(face)마다 결정한다. 지형과 카메라 시점에 따른 굴절과 반사 표현을 실시간으로 실행하기 위하여

맵을 사용한다. 유체 카툰 셰이딩 부분에서는 유체 색상톤을 기존 카툰 렌더링의 1D 텍스처에 접목하여 사용하고, 굴절 맵과 반사 맵을 사용하여 하이라이트를 비롯한 반사 굴절 효과를 표현한다.



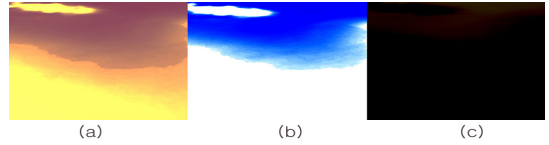
[Fig. 1] System diagram

2.2 유체 색상톤

기존의 연구에서는 유체 색상톤을 동일한 색상을 사용하여 렌더링 하지만 본 논문에서는 주변 환경을 이용하여 유체 색상톤을 생성을 위해 base texture를 정의한다. base texture를 생성하기 위해 주변 환경인 수심 빛의 입사량(광원의 위치 밝기 계수, 산란 고려)을 이용하여 유체 원색 톤을 페이스마다 결정하기 위하여 (식 1) 같이 생성한다.

$$b_i = (1 - w)v + ws \quad (\text{식 1})$$

base texture는 RGB 컬러공간(RGB color space)의 각 채널에서 수심에 따른 원색 v , 입사하는 빛의 량 s , 가중치 w 를 사용하여 원색을 만들어 준다. [Fig. 2]는 생성된 base texture로 (a)는 새벽, (b)는 낮, (c)는 밤에 대한 결과 이다.



[Fig. 2] Base texture result
(a) daybreak, (b) day, (c) night

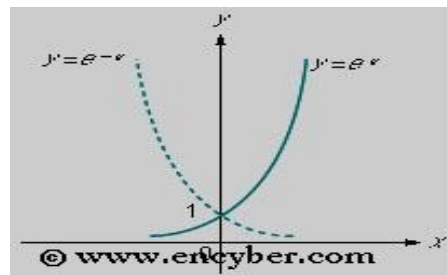
2.2.1 수심에 따른 원색

일반적으로 빛이 유체를 통과하여 흡수, 산란에 의해 유체의 색상이 나타나며, 수심이 3m가 넘어야 유체는 파란 빛을 띤다. 대체로 먼 바다의 유체의 색상은 파란색이며, 초록색, 갈색, 적갈색, 등으로 나타나는 것은 식물성 플랑크톤 때문에 나타난다[10].

본 논문에서는 수심을 이용해 RGB컬러 공간에서 색상을 표현하기 위하여 (식 2)를 이용한다.

$$v = e^{-h\lambda} \quad (\text{식 2})$$

수심에 따른 원색 표현은 수심 h 와 반감기 계수 λ 을 곱하여 지수 함수에 제공근으로 하여 표현한다. 반감기 계수 λ 는 RGB 각각 따로 가지며 값은 다르게 설정하여 생성한다. [Fig. 3]은 지수함수 그래프로 마이너스 방향으로 그래프처럼 수심에 따른 원색을 생성할 때 수심 h 앞에 마이너스 기호로 붙여 사용한다.



[Fig. 3] Exponential function graph

2.2.2 빛의 입사량에 따른 원색

일반적으로 일출, 일몰, 낮, 밤에 따라 광원의 색상이 다르게 나타나며 유체 또한 다르게 나타난다. 입사하는 빛의 양은 대기를 통과한 뒤에 나타나는 색상으로 정의하고, (식 3)의 램버트의 흡수법칙 [11]을 이용하여 생성한다.

$$s = i_o \times e^{-\mu d} \quad (\text{식 3})$$

램버트의 흡수 법칙이란 고른 매질에서 흡수 되는 정도는 빛이 투과하는 층의 두께에 비례하는 것을 기술한 법칙이다. 입사하는 빛의 양은 입사 강도 i_o 와 흡수량을 곱하여 표현된다. 흡수량은 지수함수에서의 매질의 흡수 계수 μ 에 대기권의 두께 d 를 제곱근하여 산출된다. 광원의 위치마다 강도를 따로 설정하고 매질의 흡수 계수와 대기권 두께는 동일하게 주어 낮, 밤, 새벽 등의 시간에 따라 빛의 입사량에 따른 원색을 생성하여 준다.

2.3 실시간 반사 굴절 표현

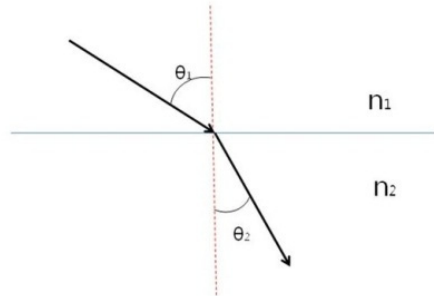
유체의 깊이감과 주변 환경을 유체에 표현하기 위해 수심이 얇은 부분에 반사 혹은 굴절을 나타낸다. 실시간 반사 굴절 표현을 위해 질감이 높지만 계산량이 많은 레이트레이싱 대신 매 프레임 (frame) 반사맵, 굴절맵을 생성하여 대략적인 물리적 효과로 주변 환경에 반사 굴절을 표현한다.

2.3.1 굴절 맵 생성

굴절 맵 생성은 한 매질에서 다른 매질로 빛이 이동하는 관계를 나타내는 스넬의 법칙(Snell's law)을 이용하여 굴절 벡터를 구할 수 있으며 굴절 맵에 굴절체를 맵에 렌더링하여 생성한다[12].

$$\frac{\eta_r}{\eta_i} = \frac{\sin\theta_i}{\sin\theta_r} \quad (\text{식 4})$$

(식 4)에 η_r 과 η_i 은 매질의 굴절률이라 하며, θ_i 는 입사각, θ_r 은 굴절각을 의미한다. 매질의 굴절률은 공기에서 1.00, 물은 1.33, 유리는 1.55이다. [Fig. 4]는 스넬의 법칙을 보여준다.



[Fig. 4] Snell's law

(식 4)을 사용하여 (식 5)를 계산 할 수 있는데 이를 사용하여 굴절 벡터와 직선이 되는 벡터를 구할 수 있다.

$$r = \frac{1}{\eta_a} i + \left(\frac{1}{\eta_a} \cos\theta_i - \sqrt{\cos^2\theta_r} \right) n \quad (\text{식 5})$$

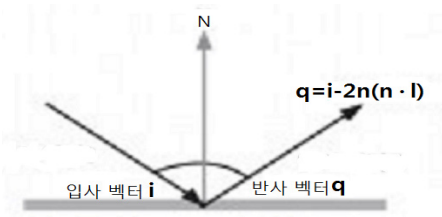
i 는 입사벡터, n 은 표면의 법선 벡터, $\eta_a = \frac{\eta_r}{\eta_i}$ 이고 r 은 굴절 벡터이다[13]. [Fig. 5]는 프로그램에서 생성된 굴절 맵의 결과이다.



[Fig. 5] Refraction map result

2.3.2 반사 맵 생성

반사맵은 입사각과 반사각의 크기가 동일하다는 반사 법칙(law of reflection)에 따라 입사 벡터와 반사 벡터가 직선이 되도록 변환 시켜 반사 맵에 반사체를 렌더링하여 반사 맵을 생성한다[14].



[Fig. 6] Reflection vector calculation

[Fig. 6]은 반사 벡터는 입사 벡터와 표면의 법선을 이용하여 계산하는 그림이고 (식 6)을 이용하여 반사 벡터를 생성할 수 있다.

$$q = i - 2n(n \cdot l) \quad (\text{식 6})$$

입사 벡터 i , 반사 벡터 q , 표면의 법선 벡터 n 일 때, 벡터간의 관계에 의해 그림과 같이 반사 벡터를 계산할 수 있다[15]. [Fig. 7]은 프로그램에서 생성된 반사 맵 결과 이다.



[Fig. 7] Reflection map result

2.4 유체 카툰 셰이딩

유체 카툰 셰이딩은 base texture에 기존 카툰

렌더링의 1D 텍스처를 접목하여 사용하며 굴절 맵과 반사 맵을 사용하여 하이라이트를 비롯한 반사 굴절 효과를 표현한다. 이를 위해 새로운 유체 카툰 셰이딩 모델을 정의 하는데 (식 7)과 같이 표현하며 이것은 기존의 블린-퐁(blinn-phong)음영 반사 모델[16]을 변경하여 생성한다.

$$i_{xy} = l_a b_i + l_d b_i o_d + (n \cdot h)^n + f \quad (\text{식 7})$$

l_a , l_d 는 빛의 주변광(ambient)과 난반사광(diffuse)계수이고, b_i 는 앞에 (식 4)에서 생성된 base texture값을 사용한다. o_d 는 불연속적인 음영 단계로 구성된 [Fig. 8]의 1차원 텍스처를 참조하여 3D 물체의 각 정점의 밝고 어두운 정도를 결정한다.



[Fig. 8] 1D texture

광원 벡터(direction vector of light)인 l 과 노멀 벡터(normal vector)인 n 을 내적(dot product)을 하여 [Fig. 8]의 텍스처 좌표 u 값으로 사용한다.

h 는 하프 벡터로 시점 벡터(direction vector of view) v 와 광원 벡터 l 의 중간의 각도에 있는 벡터로 (식 8)과 같다.

$$h = \frac{l + v}{|l + v|} \quad (\text{식 8})$$

$(n \cdot h)^n$ 은 하이라이트 표현을 하기 위해 사용되며 n 은 계수로 사용되어진다.

액체의 표면에 특징인 빛의 반사 현상과 굴절 현상 등이 나타난다. 빛의 반사와 굴절의 비율을 조정하기 위해서는 매질에서 빛의 반사와 굴절의 조정하기 위해서는 매질에서의 빛의 반사와 굴절의 F 를 계산하기 위해 주로 (식 9)와 같은 프레넬(Fresnel)방정식을 이용한다.

$$f = \frac{1}{2} \frac{\sin(\theta_2 - \theta_1)^2}{\sin(\theta_2 + \theta_1)^2} + \frac{\tan(\theta_2 - \theta_1)^2}{\tan(\theta_2 + \theta_1)^2} \quad (\text{식 9})$$

하지만 본 논문에서는 수행 속도를 높이기 위해서 계산이 복잡하고 정확한 프레넬 방정식 대신 (식 10)과 같이 단순화한 프레넬 방정식을 이용하여 액체에 대한 반사와 굴절의 표현 비율 F 를 계산한다[17].

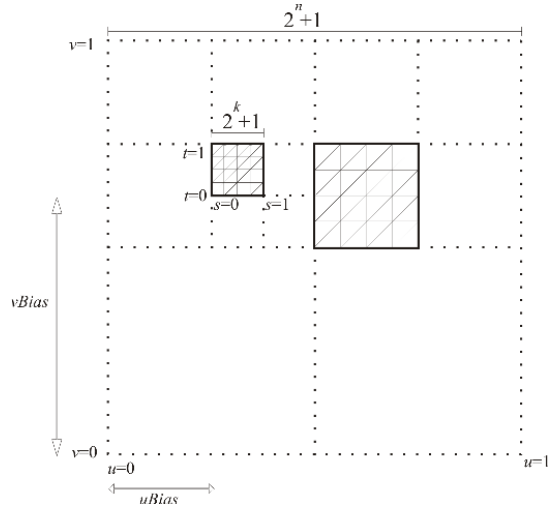
$$f = bias + size \times (1 + i \cdot np) \quad (\text{식 10})$$

단순화한 프레넬 방정식은 입사 벡터와 노멀 벡터와의 각도를 고려하여 두 벡터간의 각도에 따라 반사 굴절 비율을 나타내어 현실세계의 물리를 대략적으로 모델링 한다. $bias$ 는 프레넬 바이어스, $size$ 는 프레넬 크기, p 는 프레넬 곱수라고 한다. 이렇게 계산된 반사 계수를 사용하여 반사 맵과 굴절 맵을 조합하여 반사 굴절을 표현한다[18].

3. 실험 환경 및 결과

3.1 메쉬 생성

유체의 메쉬는 격자(grid)를 사용한다. 격자는 여러 개의 블록들로 나뉘고, 각 블록은 고정된 삼각형들로 담는다. 블록의 정점의 개수는 $(2^k + 1) \times (2^k + 1)$ 만큼 격자 형태로 생성한다. 사분트리로 격자의 LOD(Level of Detail) 제어에 사용한다. [Fig. 9]은 격자와 사분트리의 모양을 보여준다[19].



[Fig. 9] Grid and quadtree create

유체의 움직임은 정점 웨이더를 이용하여 생성한다. 유체의 파형은 wave방정식[20]인 (식 11)~(식 13)을 사용하여 정점의 위치를 조정한다.

$$x = x_0 - (k - \kappa)a \sin(k \cdot x_0 - \omega t) \quad (\text{식 11})$$

$$y = a \cos(k \cdot x_0 - \omega t) \quad (\text{식 12})$$

$$\kappa = \frac{2\pi}{\lambda} \quad (\text{식 13})$$

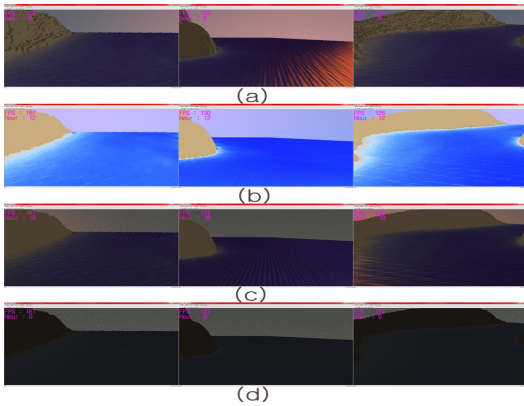
정점의 위치는 산출된 x, y 을 이용한다. 이때 a 는 진폭, t 는 시간, k 는 수평벡터, ω 는 위상, κ 는 주파수로 웨이브의 거리 λ 에 2π 를 나눠 산출된다.

3.2 실험 환경 및 결과

실험은 Intel core i5 760 2.80GHz프로세서에 4GB 주 메모리와 nVidia GeForce GTS 450을 갖는 시스템에서 수행 되었다. 그리드 크기는 1024×1024 에 평균적으로 제안된 기법의 처리 속도를 측정된 결과 약 125fps의 성능을 보여주었다.

[Fig. 10]의 (a)는 일출, (b)는 낮, (c)는 일몰, (d)는 밤으로 수심의 최고 깊이는 -156m로 설정

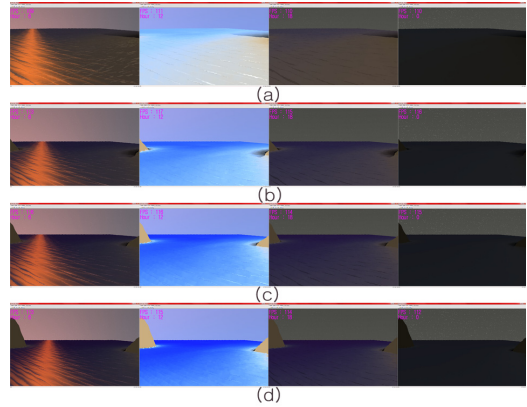
하고 시점을 변경하였고, 입사광의 강도는 (a) R:0.8 G:0.3 B:0.0, (b) R:0.7 G:0.7 B:0.7, (c) R:0.8 G:0.3 B:0.0, (d) R:0.25 G:0.25 B:0.25로 구성된 결과를 나타낸다.



[Fig. 10] View change result
(a)sunrise (b)day (c) sunset (d) night

카메라의 위치를 변경하며 나타난 결과로 광원의 반사가 나타나는 장면과 나타나지 않는 장면이 표현되고 시점에 따라 반사 굴절이 표현되어져 있고 유체 톤이 다양하게 나타난다.

[Fig. 11]은 최대 수심을 변경하여 생성된 결과로 (a)는 최고 수심 -35m, (b)는 최고 수심 -85m, (c)는 최고 수심 -135m, (d)는 최고 수심 -185m의 결과이다.



[Fig. 11] Depth of water change result
(a) 최고 수심 -35m (b) 최고 수심 -85m
(c) 최고 수심 -135m (d) 최고 수심 -185m

얕은 수심에서 굴절 표현과 투명하게 유체가 표현되어지며 수심에 따라 유체 톤이 변경되는 것을 볼 수 있다.

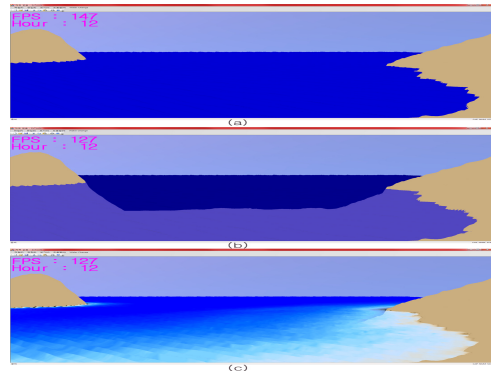
3.3 기존 유체 카툰 렌더링 방법과 비교

비교를 위해 [Fig. 12]은 일출, [Fig. 13]은 일몰, [Fig. 14]는 낮, [Fig. 15]은 밤에 대한 제안된 방법과 Eden et al.과 You et al.에 대한 카툰렌더링 방법을 구현하여 결과를 구성하였다.

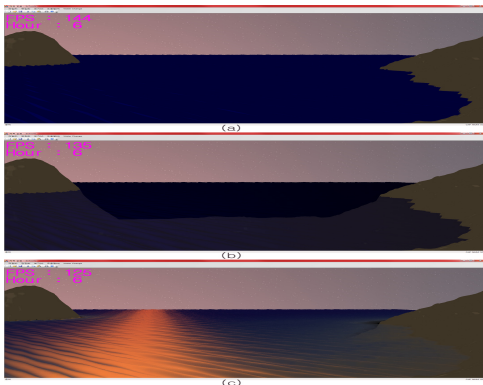
시간이 변하면 광원의 위치가 변해 명암이 달라지지만, Eden et al.의 결과는 유체의 동일한 파란색으로, You et al.의 결과는 굴절과 유체의 색상은 동일한 파란색으로 유체가 표현되었다. 하지만 제안된 방법에서는 유체 톤을 위해 base texture와 수심과 카메라 시점에 따른 실시간 반사 굴절을 표현하여 시간이 변하는 장면에서 기존의 방법보다 자연스러운 유체 톤, 유체의 깊이감, 그리고 유체의 파형이 잘 표현되었다.

제안된 방법의 결과 중 일출에서 유체 톤은 진한 파란색에서 파란색으로, 광원 반사로 과장된 일출 효과가 표현되어져 있으며 일몰에서 유체 톤은 파란색 계통으로 다양한 색상이 나타났으며, 수심이 얕은 부분에서 굴절 효과가 뚜렷하게 나타나지

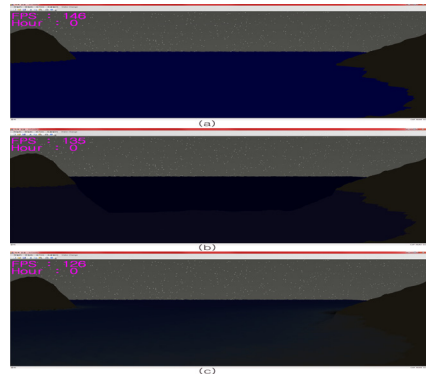
만 광원 반사가 부족한 부분이 있다. 낮에 유체 톤은 파란색에서 하늘색으로 이어지며 유체의 깊이감이 잘 표현되었고 유체의 동일한 색상으로 인해 [Fig. 14](a), (b)에서는 유체의 파형이 보이진 않지만 제안된 결과에서는 세밀하게 나타나며 밤에 유체 톤은 남색에서 진한 파란색으로 이어져 있으며 유체와 굴절 부분에 어둡게 표현되어 있고 광원 반사가 나타나지 않았다.



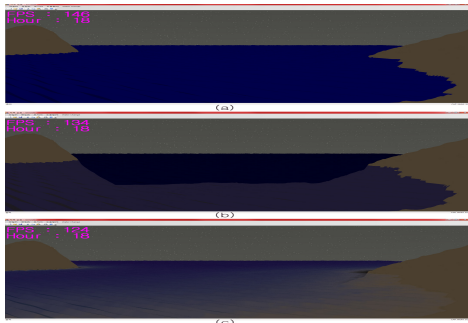
[Fig. 14] Day result (a) Eden et al. (b)You et al. (c) Proposed Method



[Fig. 12] Sunrise result (a) Eden et al. (b)You et al. (c) Proposed Method



[Fig. 15] Night result (a) Eden et al. (b)You et al. (c) Proposed Method



[Fig. 13] Sunset result (a) Eden et al. (b)You et al. (c) Proposed Method

단점으로는 유체를 렌더링 할 때 지형의 높이를 추가적으로 사용하고 매 프레임마다 base texture를 생성하여 기존의 방법보다 FPS (Frame Per Second)는 10~15%정도 낮은 수치를 보여준다.

4. 결 론

본 논문에서는 장면 설정에 어울리도록 유체를 실시간 카툰 스타일로 표현할 수 있는 실시간 방법을 제안하고자 한다. 기존의 유체를 카툰 스타일로 렌더링 하는 연구에서는 주어지는 색상 톤을 기준으로 세 개 정도의 음영단계로 메쉬의 색상을

단순화 하고, 부유 물체가 있을 때 반사 굴절을 표현하는 방식이 추가적으로 제시되었다. 본 논문에서는 광원, 수심, 지형, 시점 등에 적응적으로 유체를 카툰 스타일로 표현하는 방식을 제안한다.

유체 톤을 위해 수심에 따른 원색 톤과 입사하는 빛의 양을 이용하여 base texture를 정의하고 시간이 변함에 따라 유체 톤을 생성한다. 반사와 굴절을 실시간으로 표현하기 위해 맵을 사용하여 생성한다. base texture와 반사 굴절 맵을 이용하는 유체 카툰 셰이딩 모델을 사용하여 강한 빛의 반사로 하이라이트, 시차가 변화하는 장면에서 기존의 방법보다 자연스러운 유체톤, 수심이 얇은 부분에서의 반사 굴절을 통해 유체의 깊이감 표현이 실험을 통해 증명하였다.

차후 연구 방향으로는 강한 빛을 반사하는 방법을 변경하여 하이라이트 표현을 정밀하게 하는 것과 카툰에서 표현하는 외곽선을 이용해 유체의 곡형선을 표현하는 연구를 이어갈 예정이다.

REFERENCES

- [1] Yuehu Liu, Yuanqi Su, Yu Shao, and Daitao Jia, "A Parameterized Representation for the Cartoon Sample Space", *Advances in Multimedia Modeling, Lecture Note in Computer Science Vol.5916/2000*, pp 767~772, 2010.
- [2] McCloud, Scott, "Understanding Comics : The Invisible Art", Harper Perennial, pp290, 312 1994.
- [3] T. Akenine-Moller, E. Haines, N.Hoffman. "Real-Time Rendering 2판 7.1 Cartoon shading", infopub, pp 330~332, 2003.
- [4] A.M Eden, A.W Bargteil, T.G Gokrekin, S.B Eisinger, J.F O'Brien "A method for cartoon-style rendering of liquid animations", In *Proceedings of Graphics Interface*, pp 51~55, 2007.
- [5] M. You, J. Park, B. Choi, J. Noh, "Cartoon Animation style Rendering of water", LNCS, vol. 5875, pp 67~78, 2009.
- [6] <http://www.sealonline.co.kr/>
- [7] <http://mabinogi.nexon.com/>
- [8] <http://argo.mgame.com/>
- [9] <http://kr.battle.net/wow/ko/>
- [10] 피에르 라즐로, "바다는 왜 파랄까", 믿음IN, pp 9~24, 2006.
- [11] <http://terms.naver.com/entry.nhn?docId=754120>
- [12] T.Akenine-Moller, E.Haines, N.Hoffman, "Real-Time Rendering 2판 6.11 굴절", infopub, pp 283~286, 2003.
- [13] 최중현, 이상욱, 김용준 "좋은 게임을 만드는 핵심 원리 : 게임 수학과 물리 8.03.1 환경맵을 이용한 굴절 효과 스텔의 법칙", Hanbit Media, pp 203~205, 2003.
- [14] T.Akenine-Moller, E.Haines, N.Hoffman, "Real-Time Rendering 2판 6.10 반사", infopub, pp 276~283, 2003.
- [15] 최중현, 이상욱, 김용준 "좋은 게임을 만드는 핵심 원리 : 게임 수학과 물리 8.02.3 환경맵을 이용한 반사 효과 반사 벡터 계산", Hanbit Media, pp 201~202, 2003.
- [16] J.F. Blinn "Models of light reflection for computer synthesized pictures", SIGGRAPH vol.11, no2 1977.
- [17] R. Fernando, M. Kilgard, "The Cg Tutorial: The Definitive Guide to Programmable Real-Time Graphics", Addison-Wesley. pp 169~197 2003.
- [18] 최중현, 이상욱, 김용준 "좋은 게임을 만드는 핵심 원리 : 게임 수학과 물리 8.03.3 환경맵을 이용한 굴절 효과 프레넬 효과", Hanbit Media, pp 205~206, 2003.
- [19] H. Vistnes, "Game programming Gems6, 5.5 GPU Terrain Rendering", 와우북스, pp 533~545.
- [20] J. Tessendorf, "Simulating Ocean Water", *Simulating Nature: Realistic and Interactive Techniques. SIGGRAPH2001 Course Note*. 2001.



김 광 태 (Kwangtae Kim)

중앙대학교 첨단영상대학원 영상학과 석사과정

관심분야 : Real-Time Rendering, Game Engine



박 경 주 (Kyoungju Park)

2004년 Dept. of computer and Information Science,
University of Pennsylvania, PA

2004년-2005년 Research Professor, Rutgers
University, NJ

2007년-현재 중앙대학교 첨단영상대학원 교수

관심분야 : Physics Based Simulation,
Computer Graphics
