

변형물체 시뮬레이션을 활용한 비 마커기반 증강현실 시스템 구현[☆]

Implementation of Markerless Augmented Reality with Deformable Object Simulation

성 낙 준¹ 최 유 주² 홍 민*¹
Nak-Jun Sung Yoo-Joo Choi Min Hong

요 약

최근 제한적인 마커기반 증강현실의 여러 가지 단점들을 보완하기 위해 사용자의 얼굴, 발, 손 등을 활용한 비 마커기반 증강현실 시스템에 관한 연구들이 활발하게 진행되고 있는 추세이다. 또한 대부분의 기존 증강현실 시스템들은 사용자에게 보여주는 것과 기본적인 상호작용에 목표를 두고 강체를 증강하여 수행되는 경우가 많았다. 본 논문에서는 단지 보여주는 것에 국한되는 것이 아니라 여러 분야에서 활용이 가능한 변형물체를 사용자와의 상호작용을 바탕으로 시뮬레이션을 제공하는 비 마커기반의 증강현실 시스템을 설계 및 구현하였다. 변형물체는 질량-스프링 모델, 유한 요소 모델 두 가지 방법을 주로 사용하여 구현한다. 질량-스프링 모델은 실시간 시뮬레이션에 장점이 있으며 유한 요소 모델은 변형물체의 정밀함을 나타낼 때 장점을 가진다. 본 논문에서는 실시간으로 시뮬레이션을 목표로 하고 있기 때문에 질량-스프링 모델을 기반으로 하는 테트라헤드론 구조를 이용하여 변형물체를 구현하였다. 변형물체의 자연스러운 움직임 실시간으로 시뮬레이션하기 위해 키넥트 SDK를 통해 사용자의 손의 위치를 추적 하고, 손의 위치 변화량을 바탕으로 힘을 계산한다. 이를 바탕으로 4th order Runge-Kutta Integration 수치적분법을 이용하여 물체의 다음 위치를 계산하여 시뮬레이션 하도록 하였다. 그리고 자연스러운 동작을 표현하기 위해서 사용자의 손을 통해 물체에 작용하는 힘이 너무 많이 작용하지 않기 위해 계층에 임계값을 정하였으며 해당 임계값을 넘는 힘이 작용할 경우 임계값으로만 적용되도록 설정하였다. 각 실험을 5회씩 반복하였으며 실험에 따른 시뮬레이션 연산속도를 분석하였다. 본 논문을 통해 구현한 변형물체를 활용한 비 마커기반 증강현실 시스템을 바탕으로 기존의 강체 기반의 증강현실에서 활용하기 힘들었던 의료, 교육 및 다양한 방법으로 시뮬레이션이 가능할 것으로 기대한다.

☞ 주제어 : 변형물체 시뮬레이션, 증강현실

ABSTRACT

Recently many researches have been focused on the use of the markerless augmented reality system using face, foot, and hand of user's body to alleviate many disadvantages of the marker based augmented reality system. In addition, most existing augmented reality systems have been utilized rigid objects since they just desire to insert and to basic interaction with virtual object in the augmented reality system. In this paper, unlike restricted marker based augmented reality system with rigid objects that is based in display, we designed and implemented the markerless augmented reality system using deformable objects to apply various fields for interactive situations with a user. Generally, deformable objects can be implemented with mass-spring modeling and the finite element modeling. Mass-spring model can provide a real time simulation and finite element model can achieve more accurate simulation result in physical and mathematical view. In this paper, the proposed markerless augmented reality system utilize the mass-spring model using tetrahedron structure to provide real-time simulation result. To provide plausible simulated interaction result with deformable objects, the proposed method detects and tracks users hand with Kinect SDK and calculates the external force which is applied to the object on hand based on the position change of hand. Based on these force, 4th order Runge-Kutta Integration is applied to compute the next position of the deformable object. In addition, to prevent the generation of excessive external force by hand movement that can provide the natural behavior of deformable object, we set up the threshold value and applied this value when the hand movement is over this threshold. Each experimental test has been repeated 5 times and we analyzed the experimental result based on the computational cost of simulation. We believe that the proposed markerless augmented reality system with deformable objects can overcome the weakness of traditional marker based augmented reality system with rigid object that are not suitable to apply to other various fields including healthcare and education area.

☞ keyword : Deformable Object Simulation, Augmented Reality

¹ Dept of Computer Software Engineering, SoonChunHyang University, Asan, 336-745, Korea.

² Dept of Newmedia, Seoul Media Institute of Technology, Seoul, 157-930, Korea.

* Corresponding author (mhong@sch.ac.kr)

[Received 16 February 2016, Reviewed 24 February 2016, Accepted 9 May 2016]

☆ 본 연구는 순천대학교 학술연구비 지원으로 수행하였음

☆ 본 논문은 2015년도 인터넷정보학회 추계학술발표대회 우수논문 추천에 따라 확장 및 수정된 논문임

1. 서 론

증강현실(Augmented Reality)은 현실과 가상세계를 혼합한 것으로 현실의 영상에 가상의 물체들을 인식하여 사용자의 실제 눈에 보이지 않는 물체에 대한 감각과 인식을 높이는 역할을 하는 기술이다. 최근 사용자의 손, 머리, 발 등을 이용하여 사용자와 물체간의 상호작용이 가능한 증강현실 시스템에 관한 수요가 늘고 있고, 이에 따라 관련 기술들이 나날이 발전하고 있는 추세이다. 또한 증강현실을 편리하게 구현하기 위한 Software Developer Kit(SDK)[1]과 라이브러리 Toolkit[2]이 제공되고 있어 증강현실 기술 발전 및 확장에 기여하고 있다. 현재 증강현실은 의료, 산업, 교육 분야 등에서 다양하게 활용되고 있다. 특히 의료 분야의 경우 수술 시뮬레이션을 위해 증강현실을 사용하는 경우가 증가하고 있고, 교육 분야에서도 2차원인 책을 3차원 정보로 변환하여 학생들의 집중도를 높이기 위한 방법으로 증강현실을 사용하고 있다[3]. 이처럼 많은 분야에서 최근 증강현실 시스템의 장점을 활용하고자 다양하게 도입 및 활용하고 있는 추세이다.

그러나 현재 활용되고 있는 증강현실 기술들은 현실의 영상에 강체를 가상의 물체들로 활용하고 있고, 약속된 마커를 활용하여 물체를 증강하는 경우가 대부분이다. 하지만 의료 및 교육 분야에 대한 증강현실 시스템의 활용도를 높이기 위해서는 힘을 가해도 변형되지 않는 강체를 가상의 물체로 활용하는 것 보다 상호작용에 따라 물체의 형태가 변형되는 물체를 사용하는 것이 활용도 및 몰입도의 상승에 긍정적인 영향을 주게 된다.

따라서 본 논문에서는 여러 가지 환경에서 활용이 가능한 변형물체를 기반으로 증강현실 시스템을 설계하고 구현 한다. 또한 기존의 사용이 불편한 마커기반의 증강현실 기술을 대신하여 사용자와의 상호작용이 원활하도록 비 마커기반의 증강현실 기술을 통해 사용자의 손의 위치를 기반으로 물체를 증강하는 시스템을 구현하였다.

본 논문은 다음과 같이 구성된다. 2장에서는 관련연구 분석으로 변형물체 시뮬레이션과 기존의 증강현실 시스템에 대해 분석하고, 3장에서는 변형물체를 활용한 증강현실 시스템 설계 방법을 설명하고, 4장에서는 변형물체를 활용한 증강현실 시스템 구현 및 실험 결과를 소개하고, 5장에서는 본 논문의 결론을 기술하도록 한다.

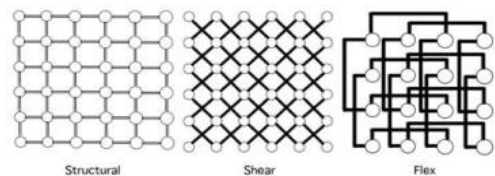
2. 관련연구

2.1 변형물체 시뮬레이션

물리 기반의 시뮬레이션은 관련 기술의 발전으로 꾸

준하게 연산속도 및 시뮬레이션 수행에 대한 결과의 안정성이 향상되고 있다. 어떠한 힘을 가해도 모양이 변하지 않는 강체에 비해 활용도가 높아지고 있는 변형물체의 경우 지속적인 연구가 수행되고 있다. 또한 변형물체는 최근 여러 가지 엔진들을 통한 시뮬레이션 결과들이 제공되고 있는데 대표적인 엔진으로 Bullet, Carbon, Cry, Intel Havok Cloth, NVIDIA PhysX 등이 있다.

이러한 변형 물체 시뮬레이션을 표현하는 방식은 크게 두 가지로 분류할 수 있다. 연산 속도는 느리지만 물체 변형의 정밀함을 나타내기 위한 유한 요소 방법(Finite Element Method), 연산 속도가 빨라 실시간으로 시뮬레이션을 하기 위한 질량-스프링 모델(Mass-Spring Model) 방법이 널리 사용되고 있다. 이러한 방법들은 변형물체 시뮬레이션들이 크게 적용되는 두 가지 방법이며, 이러한 두 가지 모델을 통하여 상황에 맞게 표현 및 시뮬레이션하고 있다. 실시간으로 시뮬레이션을 수행할 때 가장 많이 적용되고 있는 방법인 Mass-Spring 모델 방법은 질량을 가진 노드들이 질량이 없는 가상의 스프링으로 연결된 구조를 가진다. 가상의 스프링은 연결되는 구조에 따라 Shear, Flex(Bend), Structural 스프링으로 표현된다. Structural 스프링은 노드와 인접한 가로, 세로의 노드를 연결하는 스프링으로 퍼질 때 힘을 전달하며, Shear 스프링의 경우 전단 응력(접하는 방향으로 작용하는 힘)이 작용하였을 때 힘을 전달하는 역할을 가진다. Flex(Bend) 스프링은 굴곡 응력이 작용하였을 때 힘을 전달하는 역할을 하는 스프링이다[4]. 따라서 Mass-Spring 모델은 어떠한 부분에 가해진 힘이 연결된 스프링들을 통해서 물체의 다른 부분들로 전파되어 영향을 받게 된다[4-6].



(그림 1) Mass-Spring 시스템의 스프링 유형
(Figure 1) Spring types in Mass-Spring System

한 개의 가상 스프링은 다음과 같은 정보들로 구성된다. 스프링은 스프링 계수(K_s), 복원 계수(K_d), 스프링 초기 길이(L_0), 연결된 두 개의 노드의 정보로 구성되며 이러한 정보를 바탕으로 후의 공식(Hook's law)을 통해 스프링에 연결된 두 개의 노드에 작용하는 힘을 계산할

수 있다. 이를 통해 계산한 힘의 값은 수치 적분을 통해 일정한 시간 간격 뒤의 노드 위치 정보를 구하는데 사용한다. 수치적분은 다음 식인 뉴턴의 운동 제 2법칙을 이용해 계산한다.

$$f = m \times a \text{ or } \ddot{x} = \frac{f}{m} \quad (\text{식1})$$

2.1.1 수치적분

뉴턴의 운동 제2법칙을 통해 구한 힘을 통해 일정 시간 뒤의 속력을 구할 수 있다. 또한 그 속력을 바탕으로 일정 시간 뒤의 위치를 예측할 수 있다. 이는 수치적분을 통해 계산하게 되며, 수치적분은 적분을 이용하여 일정 시간 뒤의 상태의 값을 도출해내는 방법이다. 일정 시간은 Δt 로 표현하며 기준이 되는 현재 시간 t_1 에서 물체의 위치, 속도를 이미 알고 있으며 힘이 시간, 위치, 속도와 관계함수이면 $t_2 = t_1 + \Delta t$ 일 때의 속도 및 위치를 예측하여 계산하는 것이 가능하다. 보통의 경우 수치적분은 Euler Integration, Midpoint Integration, 4th order Runge-Kutta Integration 방법 등을 통해 수행된다.

2.2 증강현실 기술

증강현실은 대표적으로 마커(식별자, Marker)기반방식과 비 마커(비식별자, Markerless(Non-Marker)) 기반방식으로 구분할 수 있다. 마커기반 증강현실은 증강할 위치에 부착된 마커를 이용하여 가상의 3차원 물체를 증강하는 방법을 말하며 카메라 영상과 미리 약속된 마커만으로 구현이 가능하고 마커에 대한 인식이 간단해 실시간으로 활용할 수 있다는 장점을 가진다. 하지만 조명에 의존적이며 물체를 증강하기 위해서는 항상 약속 및 학습된 마커가 영상에 존재해야하는 의존적인 단점을 가지고 있다. 그림2는 마커기반 증강현실의 예시이다. Prince Lab에서 초등생 교육용으로 제작한 과학 콘텐츠인 Science Fun의 실행화면으로 지구 내부의 모습을 나타낸 구체가 마커를 기반으로 증강된 모습을 확인할 수 있다[7].



(그림 2) 마커 기반 AR 시스템의 예
(Figure 2) Example of marker based AR system

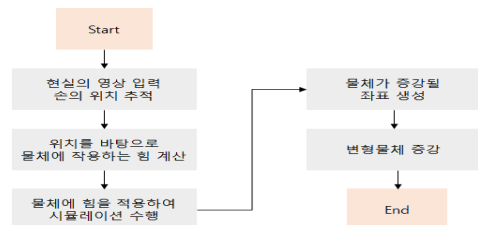
대표적인 마커기반 증강현실 기술 관련 라이브러리는 ARToolkit, ARToolkitplus 등이 있다. 위와 같은 마커기반 증강현실 기술은 등록된 마커를 기반으로 가상의 물체를 증강하기 때문에 약속된 형태의 마커에 의존되는 단점을 가진다. 이를 보완하기 위해 비 마커기반 증강현실 기술이 개발되었고 이와 더불어 모바일 환경에서의 증강현실 기술 또한 연구가 활발히 진행되고 있는 추세이다[8,9]. 비 마커기반 증강현실 기술이란 마커에 의존하는 것이 아니라 입력된 영상에서 증강될 3차원 물체가 위치할 좌표만 설정하게 되면 자연스럽게 물체를 증강시킬 수 있는 기술이다. 하지만 물체를 증강시키기 위해 해야 할 계산의 양이 마커기반 증강현실보다 많고 구현 난이도 또한 어려운 단점이 있다.

이러한 비 마커기반 증강현실 기술 중 사람의 손을 이용하여 가상의 3차원 물체를 증강시키고 상호작용하는 연구들이 꾸준히 진행되고 있다[10,11]. 이처럼 손을 이용한 비 마커기반의 증강현실 기술들은 사용자의 몰입감을 증가시키고 사용자와 가상 물체와의 상호작용이 기존의 마커기반 증강현실보다 더 편리하고 몰입감을 높이는 장점을 가지고 있다. 그러나 계속해서 손의 위치를 추적하고 증강좌표계를 계산하면서 생기는 연산의 증가와 손이 다른 배경 혹은 물체에 가려졌을 때의 손 가림 문제 등 여러 가지 단점들을 가지고 있다.

3. 변형물체 시뮬레이션을 활용한 증강현실 시스템 설계

3.1 시스템 흐름도

본 논문은 실시간으로 입력되는 현실 영상으로부터 왼손의 위치를 추적하고 그 위치를 바탕으로 3차원 물체를 증강하는 비 마커기반의 증강현실 시스템을 설계 및 구현하였다. 다음 그림3은 본 논문에서 설계 및 구현한 증강현실 시스템의 흐름도이다.



(그림 3) 제안된 증강현실 시스템 흐름도
(Figure 3) Flowchart of proposed AR system

증강현실 시스템의 기본 구성요소인 현실의 영상을 입력받기 위해 본 논문에서는 마이크로소프트사의 키넥트 v2를 사용하였으며, Kinect SDK를 사용하여 왼손의 위치를 실시간으로 추적 하였다[12,13]. 실시간으로 사용자의 왼손의 위치와 상태를 추적하고 time-step별 왼손의 위치 변화량을 계산한다. 물체가 손위에 있을 때의 왼손의 위치의 변화량은 변형물체에 외부 힘으로 적용되어 시물레이션을 수행하였다. 실시간으로 시물레이션을 수행함과 동시에 변형물체가 증강될 좌표를 손의 위치로부터 얻어 생성하고 증강하였다.

4. 변형물체 시물레이션을 활용한 증강현실 시스템 구현 및 실험 결과

4.1 변형물체 시물레이션

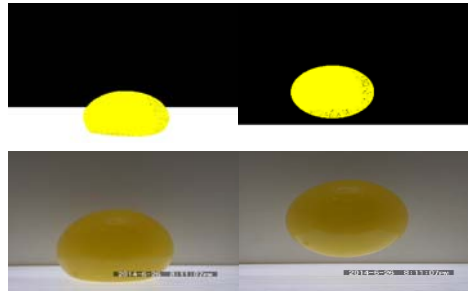
본 논문에서는 변형물체를 구현하기 위해 질량-스프링 모델을 기반으로 하는 테트라헤드론 구조의 3D 입체 변형물체 모델을 설계 및 모델링 하였다. 본 논문에서 적용된 변형물체 내부의 모든 노드들은 테트라헤드론 구조를 이용하여 구체를 설계하였다. 모든 정점을 스프링으로 연결한 메쉬의 형태를 가지며 내부의 정점 또한 모두 연결되어 있어 변형물체를 구현하기 용이한 장점이 있다. 본 논문에서 사용한 구체 모델은 테트라헤드론 모델링 프로그램인 Tetgen을 통하여 모델링 하였다[14]. 표 1은 Tetgen을 통해 생성한 메쉬 모델들의 정보이다. 총 세 가지 메쉬 모델을 생성하였다. 모델의 이름은 연결된 스프링의 개수를 바탕으로 설정하였으며 모델은 스프링, 노드, 노드의 위치, 외부 힘 등의 정보를 가진다.

(표 1) 메쉬 모델 정보
(Table 1) Mash model information

구분	spring 수	node 수	face 수	Ks	Kd
ele 1000	1,483	241	200	350	0.01
ele 3000	3,042	612	1,048	600	0.01
ele 6000	6,481	1,270	2,120	970	0.02

다음 그림4는 본 시스템에서 구현한 테트라헤드론 구조를 가지는 변형물체를 직접 시물레이션한 영상과 실제 떨어지는 공을 초고속 카메라로 촬영한 영상을 비교한 그림이다. 본 실험을 위해 ele 6000 모델을 사용하였으며 바닥과 충돌했을 때 물체가 눌렀다가 복원되는 모습을

확인할 수 있다. 또한 충돌 후의 모습 또한 현실감 있게 변형되는 것을 확인할 수 있다. 이를 시물레이션하기 위해 4th order Runge-Kutta 수치적분법을 이용하여 연산의 속도는 떨어지지만 실제와 가장 유사한 결과를 나타내도록 구현하였으며 식2를 통해 스프링에 작용하는 힘을 계산하였다.



(그림 4) 실제 카메라와 시물레이션 결과 이미지 비교
(Figure 4) Comparison between real camera image and simulated result image

$$\begin{aligned}
 dir &= node1.pos - node2.pos \\
 length &= dir.size \\
 spforce &= direction \times K_s \times (length - L_0) \quad (식3) \\
 damp &= (node2.vel - node1.vel) \times K_d \times t
 \end{aligned}$$

스프링에 연결된 노드들의 위치 차를 통해 방향(*direction*)을 설정하고 위치 차의 길이(*length*)를 계산한다. 계산한 두 가지 정보와 스프링이 가지는 기본 정보인 스프링계수(*Ks*), 복원계수(*Kd*), 스프링 초기길이(*L0*) 값을 통해 스프링에 작용하는 힘(*spforce*)과 복원 값(*damp*)을 계산한다. 이후 하나의 노드에는 힘이 작용하고 다른 하나의 노드에는 힘이 반작용하기 때문에 식3과 같이 스프링에 연결된 노드에 *spforce* 값을 연산하여 적용하였다.

$$\begin{aligned}
 spforce &+= damp \\
 node1.force &+= spforce \\
 node2.force &-= spforce
 \end{aligned} \quad (식3)$$

4.2 손의 위치 추적

본 논문에서 변형물체를 증강할 좌표를 설정하고 다이내믹 시물레이션을 수행하기 위한 외부의 힘을 설정하기 위해서는 손의 정확한 위치 추적이 중요하다. 기존에 키

넥트에서 입력받은 영상을 통해 손 영역을 검출하는 연구들이 많이 진행되어 왔다[15]. 본 논문에서는 보다 쉽게 손 영역을 검출할 수 있도록 키넥트 SDK의 Hand-Tracking 방법을 사용하여 사용자의 손 Joint의 위치를 실시간으로 추적하였다. 이 때 키넥트 SDK를 통해 얻을 수 있는 정보는 손의 상태(open, close, not detect)와 손의 위치정보(X, Y, Z)가 있다. 사용자가 가상의 물체와의 상호 작용하기 위해서는 손에 관한 여러 가지 제스처를 인지하여야하기 때문에 손의 여러 가지 상태를 추정하여 저장하는 것 또한 중요하다. 다음 그림5는 본 논문에서 사용한 키넥트 SDK의 Hand Tracking의 결과 화면이다.



(그림 5) 키넥트 SDK를 사용한 손 감지
(Figure 5) Hand detection result using Kinect SDK

실시간으로 Left Hand와 Right Hand의 Open, Close 상태뿐만 아니라 손의 위치를 측정하여 저장한다. Hand Tracking을 통해 측정된 위치 값은 (-1, -1)~(1, 1) 사이의 정규좌표로 측정되며 저장된 값을 통해 최초로 물체를 증강할 위치와 물체가 놓인 손의 움직임에 따라 물체에 적용할 힘을 계산하며 수식은 다음과 같다.

$$\begin{aligned} KinectXpos &= depth.x \times HandPos.x \times 34 \\ KinectYpos &= depth.y \times HandPos.y \times 79 \quad (\text{식4}) \\ KinectZpos &= (depth.z + 1215.5) / 6.1 \end{aligned}$$

위의 수식을 사용해 영상에서 얻은 손의 위치 값을 물체가 증강될 좌표계로 설정된다. 물체가 증강된 후 손의 위치 변화량은 다음과 같은 수식을 통해 물체에 작용하는 외부 힘으로 변환한다.

$$exforce = now - previous \quad (\text{식5})$$

식5와 같이 외부 힘(exforce)을 구하기 위해 현재 왼손의 위치(now), 이전의 왼손의 위치(previous)를 계산한다. 이때 exforce, now, previous 값들은 X, Y, Z 세 축의 힘을 모두 계산하기 위해 모두 Eigen 라이브러리를

통해 사용할 수 있는 벡터 자료형으로 설정하였다[16]. now, previous 의 두 개 값은 모두 위에서 계산한 KinectPos 값을 이용하였다. 계산한 exforce 값은 물체에 작용하는 외부 힘으로 적용시킨다. 외부 힘은 변형물체가 증강된 후 사용자가 손을 빠르게 움직일수록 더 크게 계산되며 임계값을 설정하여 사용자의 제스처에 제한을 두었다. 위치 변화량이 설정한 임계값을 넘었을 경우 물체가 손에서 떨어져 주어진 방향으로 임계값으로만 적용되도록 설정하였다. 표2는 본 논문에서 구현한 시스템에서 설정한 제스처 임계값이다.

(표 2) 제스처 임계값
(Table 2) Threshold values for gestures

손의 상태	방향	임계값
Open	up	35.0
Open	left	35.0
Open	right	35.0
Close	up	45.0
Close	left	45.0
Close	right	45.0

키넥트를 통해 얻을 수 있는 또 다른 정보로 사용자의 손의 깊이정보가 있다. 본 시스템은 사용자의 손의 깊이정보를 바탕으로 가상의 변형물체의 크기를 조절하기 위해 앞서 계산한 KinectZpos 값에 따라 near, middle, far 세 가지로 손의 깊이 상태를 분류하였다. 키넥트에서 손의 위치가 멀어질수록 KinectZpos 값이 작아지는 것을 바탕으로 분류하였으며, 다음 그림6과 같은 결과를 나타낸다.



(그림 6) 손의 위치에 따른 깊이 분류
(Figure 6) Depth classification with hand position

4.3 실험 환경 및 결과

본 논문에서 제안한 시스템은 PC를 기반으로 구현한 시스템이다. 시스템을 수행하기 위해 설정된 환경은 다음 표3과 같다. 현실의 영상을 입력받는 장비인 마이크로소

프트 키넥트 v2를 사용하기 위한 최소 환경인 Windows 8.1과 USB 3.0 지원을 만족하는 환경을 구성하였다. 앞서 설명한 손의 위치 추적 및 손의 깊이 분류를 본 시스템에 활용하였으며 이를 바탕으로 비 마커기반 증강현실 시스템의 기본 구조를 구현하였다. 또한 실험에 사용한 변형 물체는 앞서 모델링한 ele 1000, ele 3000, ele 6000 세 가지 모델을 사용하였으며 시물레이션에 필요한 수학적 연산을 수행하고 데이터 타입을 편리하게 설정하기 위해 행렬 라이브러리 Eigen을 사용하였다. 또한 시물레이션을 수행하기 위한 수치적분법으로 연산시간이 증가하여도 보다 정확한 시물레이션을 위해 4th order Runge-Kutta Integration을 이용하여 다음 시간의 위치를 계산하였다. 해당 방법을 이용해 시물레이션을 수행한 후 변형물체를 렌더링하기 위해 OpenGL 1.0 라이브러리를 사용하였다. 본 논문에서 수행한 실험의 설정은 다음 표4와 같다. 물체의 수와 모델을 통해 총 6가지 경우로 분류하였으며, 실험 번호 별 총 5번의 시물레이션을 통해 각각의 FPS(Frame Per Second)를 측정 및 비교하였다.

(표 3) 시스템 수행 환경

(Table 3) Performed system environment

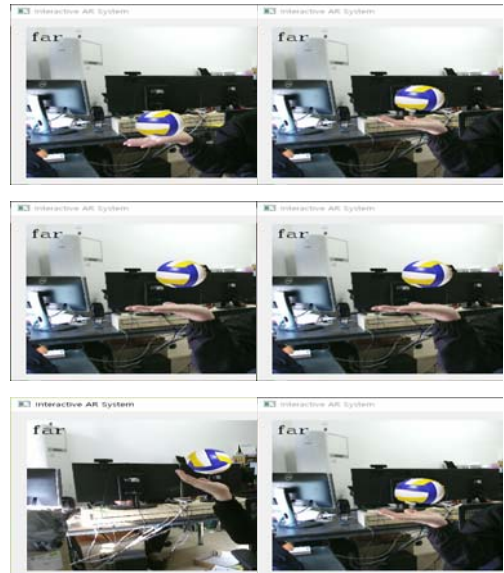
구분	세부 환경
CPU	Intel i7-3770K
RAM	16.00GB
OS	Windows 10 Pro
GPU	NVIDIA GeForce GTX 760
Video Input	Microsoft Kinect v2
Language & Library	C++, OpenGL 1.0, Eigen

(표 4) 실험 설정

(Table 4) Experimental settings

No	사용자 설정	물체의 수	모델 명
1	왼손	1	ele 1000
2	양손	2	ele 1000
3	왼손	1	ele 3000
4	양손	2	ele 3000
5	왼손	1	ele 6000
6	양손	2	ele 6000

다음은 실험들의 실행 결과 영상이다. 물체를 손의 위치에 증강한 뒤 사용자의 손과 상호작용하는 모습들을 나타낸 사진이다. 변형물체를 한 개 증강하였을 때는 왼손을 통해서만 힘을 줄 수 있으며 물체를 두 개를 증강하였을 때는 양손의 힘을 모두 변형물체에 작용할 수 있도록 구현하였다.



(그림 7) 실험 결과(물체 한 개)

(Figure 7) Experimental result with a ball



(그림 8) 실험 결과(물체 두 개)

(Figure 8) Experimental result with two balls

그림7은 왼손과의 상호작용을 통해 변형물체 한 개를 위로 던졌을 때를 나타내는 영상이고 그림8은 양손과의 상호작용을 통해 변형물체 두 개를 포물선을 그리게 던진 영상이다. 실험을 총 5번의 반복한 결과 다음 표5와 같은 FPS 차이를 확인할 수 있었다. node와 spring의 수가 적을수록 시물레이션에 최적화된 환경임을 확인할 수 있으며 node와 spring의 수가 늘어날수록 노드와 스프링에 대한 연산의 양이 많아져 현실적인 움직임과 모양을 나타내지만 실시간으로 시물레이션하기에는 적합하지 않

은 환경이 될 수 있다. 따라서 비 마커기반의 효율적인 실시간 시뮬레이션을 포함하는 증강현실 시스템의 구현을 위해서는 적절한 노드와 스프링의 수로 변형물체들의 모델링을 최적화 하는 작업이 필요하다.

(표 5) 실험 결과
(Table 5) Computational cost for examples

구분	평균 FPS
실험1	59.96 fps
실험2	58.77 fps
실험3	53.26 fps
실험4	51.17 fps
실험5	47.02 fps
실험6	37.49 fps

5. 결 론

본 논문은 마이크로소프트사의 키넥트 v2를 기반으로 하여 변형물체를 사용하는 비 마커기반의 증강현실 시스템을 설계 및 구현하였다. 시스템 실험 결과 사용자와의 상호작용을 통해 여러 가지 간단한 행동을 성공적으로 수행할 수 있는 것을 확인하였으며 증강하는 변형물체의 해상도에 따라 시스템의 성능이 변하는 것을 확인하였다.

추후 구체 뿐 아니라 다양한 물체를 증강하는 실험을 통해 교육 등의 방면으로 확장할 수 있을 것으로 예상된다. 또한 의도하지 않은 시뮬레이션 시스템의 Blow-up 현상을 방지하기 위해서는 Ks , Kd 값을 최적화하는 것이 중요하며 Mass-Spring 모델 기반의 변형물체가 아니라 유한요소방법으로의 구현을 통한 방법도 중요할 것으로 생각한다. 또한 CPU에서 변형물체 시뮬레이션을 수행하지 않고 GPU를 통해 시뮬레이션을 수행한다면 다음 위치를 구하는 수치적분의 연산 속도를 증가시킬 수 있을 것으로 예상되며 이와 관련 연구를 수행할 계획이다. 이러한 점을 바탕으로 기존의 강체만을 사용한 제한적인 증강현실 시스템에서 의료, 교육 등 여러 분야로 더 활용 및 발전 가능성이 있는 변형물체 기반의 증강현실 시스템으로 발전될 것으로 기대한다.

참 고 문 헌 (Reference)

- [1] Vuforia Developer, <https://developer.vuforia.com>
- [2] AR Toolkit, <http://www.hitl.washington.edu/artoolkit/>
- [3] Korea Communications Agency, "Augmented Reality (AR) technology and services, the latest trends and forecasts", Information Communication Technology Issues & Outlook, No. 9, 2013.
- [4] Xavier provot, "Deformation Constraints in a Mass-Spring Model to Describe Rigid Cloth Behavior", Graphics Interface, pp.147-155, 1995.
- [5] Vassilev, Tzvetomir, Bernhard Spanlang, "A mass-spring model for realtime deformable solids", Proceedings of the East-West Vision, pp.149-154, 2002.
- [6] Baraff, David, Andrew Witkin, "Large Steps in Cloth Simulation", COMPUTER GRAPHICS Proceedings, Annual Conference Series, pp.19-24, 1998.
<http://dx.doi.org/10.1145/280814.280821>
- [7] Science Fun, princelab, <http://www.princelab.co.kr/>
- [8] Junchul Chun, "Vision-based Motion Control for the Immersive Interaction with a Mobile Augmented Reality Object", JICS 2011, Vol.12, No.3, pp.119-129, 2011.
- [9] Seonho Lee, Junchul Chun, "Dynamic Behavior Modelling of Augmented Objects with Haptic Interaction", JICS 2014, Vol.15, No.1, pp.171-178, 2014.
<http://dx.doi.org/10.7472/jksii.2014.15.1.171>
- [10] Lee, T. and Hollerer, T., "Hand AR: Markerless Inspection of Augmented Reality Objects Using Fingertip Tracking", In Proceedings of IEEE International Symposium on Wearable Computers, pp. 83-90, 2007.
<http://dx.doi.org/10.1109/iswc.2007.4373785>
- [11] Lee, T. and Hollerer, T., "Hybrid Feature Tracking and User Interaction for Markerless Augment Reality", In IEEE Int'l Conference on Virtual Reality, pp. 145-152, 2008.
<http://dx.doi.org/10.1109/vr.2008.4480766>
- [12] Frati, V., Prattichizzo, D., "Using Kinect for Hand Tracking And Rendering in Wearable Haptics", World Haptic Conference IEEE, pp.317 - 321, 2011.
<http://dx.doi.org/10.1109/whc.2011.5945505>
- [13] Microsoft kinect v2 software development kit reference, <https://dev.windows.com/en-us/kinect>
- [14] Tetgen, <http://wias-berlin.de/software/tetgen/>

[15] Hanhoon Park, Junyeong Choi, Jong-Il Park, Kwang-Seok Moon, "A Study on Hand Region Detection for Kinect-Based Hand Shape Recognition", Journal of broadcast Engineering, Vol. 18, No. 3, pp.393-400, 2013.

<http://dx.doi.org/10.5909/jbe.2013.18.3.393>

[16] Eigen Library, <http://eigen.tuxfamily.org/dox/>

● 저 자 소 개 ●



성 낙 준 (Nak-jun Sung)

2012년 ~ 현재 순천향대학교 컴퓨터 소프트웨어 공학과(공학사)

관심분야 : 다이나믹 시물레이션, 변형물체 시물레이션, 증강현실, 가상현실

E-mail : njsung@sch.ac.kr



최 유 주 (Yoo-joo Choi)

1989년 이화여자대학교 전자계산학과(이학사)

1991년 이화여자대학교 일반대학원 전자계산학과(이학석사)

2005년 이화여자대학교 과학기술대학원 컴퓨터공학과(공학박사)

1991년 ~ 1993년 한국컴퓨터주식회사 기술연구소 주임연구원

1994년 ~ 1999년 포스데이타주식회사 기술연구소 주임연구원

2005년 ~ 2010년 서울벤처대학원대학교 컴퓨터응용기술학과 조교수

2010년 ~ 현재 서울미디어대학원대학교 뉴미디어콘텐츠학과 교수

관심분야 : 컴퓨터 그래픽스, 모바일 증강현실, 영상인식, 영상보정

E-mail : yjchoi@smit.ac.kr



홍 민 (Min Hong)

1995년 순천향대학교 전산학과(공학사)

2001년 University of Colorado at Boulder(공학석사)

2005년 University of Colorado at Denver(이학박사)

2006년~현재 순천향대학교 컴퓨터소프트웨어공학과 교수

관심분야 : 컴퓨터그래픽스, 다이나믹 시물레이션, 바이오 인포매틱스, 영상처리

E-mail : mhong@sch.ac.kr