

Leap Motion Framework for Juggling Motion According to User Motion in Virtual Environment

Jong-Hyun Kim*

*Professor, School of Software Application, Kangnam University, Gyeonggi-do, Korea

[Abstract]

In this paper, we propose a new framework that calculates the user's hand motions using a Leap Motion device, and uses this to practice and analyze arm muscles as well as juggling motions. The proposed method can map the movement of the ball in a virtual environment according to the user's hand motions in real time, and analyze the amount of exercise by visualizing the relaxation and contraction of the muscles. The proposed framework consists of three main parts : 1) It tracks the user's hand position with the Leap Motion device. 2) As with juggling, the action pattern of the user throwing the ball is defined as an event. 3) We propose a parabola-based particle method to map the movement of a juggling shape to a ball based on the user's hand position. As a result, using the our framework, it is possible to play a juggling game in real-time.

▶ **Key words:** Leap Motion device, Hand motion, Arm muscles, Virtual environment, Juggling motion

[요 약]

본 논문에서는 립모션 디바이스를 이용하여 사용자의 손 움직임을 계산하고, 이로부터 저글링 동작뿐만 아니라 이것을 활용하여 팔 근육을 연습 및 분석할 수 있는 새로운 프레임워크를 제안한다. 제안하는 방법은 실시간으로 사용자의 손동작에 맞춰 가상환경에서 공의 움직임을 매핑 할 수 있고, 근육의 이완과 수축을 시각화하여 운동량을 분석할 수 있다. 제안하는 프레임워크는 크게 세 부분으로 구성된다 : 1) 립모션 디바이스를 이용하여 사용자의 손 위치를 추적한다. 2) 저글링을 하듯이, 사용자가 공을 던지는 행동 패턴을 이벤트로 정의한다. 3) 사용자의 손 위치를 기준으로 저글링 형태의 움직임을 공에 매핑하기 위한 포물선 기반 입자 기법을 제안한다. 결과적으로 본 논문의 프레임워크를 이용하면 실시간 저글링 게임을 할 수 있다.

▶ **주제어:** 립모션 디바이스, 손 동작, 팔 근육, 가상환경, 저글링 모션

I. Introduction

인체의 복잡한 움직임은 근골격계의 기하학적 구조에 영향을 받으며, 인체의 골격은 몸을 지지하고, 이 골격의 움직임에 따라 근육의 이완과 수축이 표현된다[1-3]. 이러한 사용자 근육을 이용한 물리 기반 접근법들은 다양한 캐릭터의 자세, 움직임 제어 등 다양한 방면으로 발전되어 왔다[4-7]. 컴퓨터그래픽스의 응용 분야에서는 근골격계 기반의 역학을 활용하여 자연스러운 인체 자세와 움직임을 재현하려는 방법들이 꾸준히 연구되어 왔다[8]. 그러나 이러한 접근법을 VR(Virtual Reality, 가상현실)과 AR(Augmented Reality, 증강현실)같은 유비쿼터스 컴퓨팅 환경에 적용하려면 실시간으로 사용자의 팔 움직임을 추적할 수 있어야 한다.

최근에 VR 장비는 시각적인 요소뿐만 아니라 몰입감을 극대화하기 위해 전신슈트를 입는 경우도 있다[9]. 하지만 이러한 기술은 영화와 같이 고품질 모션 데이터를 얻기 위함이 아니라면, 일반 사용자들은 활용하기 어려운 시스템이기 때문에 본 논문에서는 깊이 카메라로 손의 움직임을 간단하게 예측할 수 있는 디바이스인 립모션을 활용한다[10]. 이 외에 손의 움직임만으로 팔 근육의 이완과 수축을 예측해야 되기 때문에 앞에서 언급한 문제들은 반드시 해결되어야 한다.

저글링과 같은 팔의 움직임은 반복적인 행동 패턴이 나타나며, Choi 등은 이러한 반복적인 행동 패턴을 기반으로 상호작용에 대한 규칙을 생성했고, 이것을 데이터 기반으로 합성하여 저글링 움직임을 표현할 수 있는 기법을 제안했다[11]. 이 접근법은 본 논문과 유사하게 실제 객체를 사용하는 것이 아닌, 사용자의 움직임을 분석하여 저글링 동작을 합성하는 방법이다. 하지만, 고정된 위치에서 팔의 회전만을 고려하기 때문에 동작 자체가 제한적일 수밖에 없고, 키넥트를 기반으로 움직이기 때문에 전체 골격이 정확하게 추적되어야 팔목과 같은 움직임을 계산할 수 있기에 안정성 문제가 대두될 가능성이 크다.

사람의 근육을 물리 기반으로 풀어낸 Hill-type 근육 모델은 근육의 비선형 수축 역학을 기반으로 컴퓨터그래픽스, 인체 공학, 로봇틱스 분야 등 다양한 학문에서 활용되고 있다[12]. 그러나 계산량이 크기 때문에 실시간 처리가 어렵고, 혼합현실이나 인터랙티브한 시스템으로는 활용하기 어려운 접근법이다. 또한, Hill-type 근육 모델의 Pennation 각도, 최대 힘, Way-point와 같은 다양한 물리적 파라미터들은 실제 근육의 움직임에 따라 개별적으로 조정되어야 하기 때문에 실제 사용자가 원하는 결과를 만들려면 복잡한 파라미터 튜닝과정을 거쳐야 한다.

이전 연구들에서는 골격계와 근육계 시뮬레이션들이 밀접하게 통합되어 다양한 시뮬레이션 및 제어 기법들로 확장되었지만[13], 사용자의 움직임에 맞게 객체를 제어하는 인터랙티브 시스템으로 확장하기에는 충분하지 않았다. Lee 등은 다면체와 같은 볼륨 구조와 세그먼트 컨트롤러를 이용하여 골격에 의한 근육의 움직임을 표현할 수 있는 근골격 시뮬레이션 기법을 제안했다[13]. Si 등이 제안한 체적 시뮬레이션은 주로 시각화를 위한 방법이며[14], 이 역시 FEM(Finite Element Method)과 유사하게 세그먼트 컨트롤러를 사용하여 골격과 근육에 대한 움직임을 표현했다. Fan 등은 체적 프리미티브의 동작을 기반으로 골격의 움직임을 안정적으로 표현했다[15]. 그럼에도 불구하고 제어 시스템에서 Si 등의 방법과[14] 동일한 문제점이 발견되었다. 본 논문에서는 손의 움직임을 실시간으로 분석하여 규칙 기반 합성이 아닌, 물리 기반으로 저글링 모션을 만들 수 있는 프레임워크를 제안한다.

II. Preliminaries

1. Related works

그래프 기반 방법은 데이터 기반 모션 합성 연구의 초기 단계에서 널리 사용되는 접근법이다[16]. 사용자의 모션 규격에 따라 미리 선택된 전환 지점에서만 예제 모션 클립 간의 모션 전환을 허용하는 이러한 방법은 끊임없이 변화하는 환경에 대한 반응성이 높은 모션의 온라인 합성에는 사용하기 어렵다. 최근에 실시간 모션 합성을 위한 학습 기반 접근법이 제안되었다[17,18]. 그러나 온라인 방식으로 공과 같이 움직이는 외부 물체를 처리하기 위한 학습 방법으로 확장하기는 어렵다. 본 논문에서는 사용자의 손 움직임 측정 기술에 기반하여[19], 공의 모션을 매칭하는 방식을 제안한다. 학습 기반 접근 방식과 달리, 제안하는 방법은 무거운 전처리 없이 저글링과 같은 동작을 부드럽게 합성할 뿐만 아니라, 공을 잡거나 더 세게 던지는 동작 전환을 가능하게 한다.

컴퓨터그래픽스 분야에서 물리 기반 모션 제어는 몇 년 동안 집중적으로 연구되어 왔다 : 보행식 모션뿐만 아니라 [20], 비 보행식 모션인 무술 발차기, 기어가기, 점프, 댄스 등과 같은 동작도 다루었다[21,22]. 일반적으로 전신 운동 제어를 위한 역학 방정식은 환경과의 접촉점 변화로 인한 불연속으로 인해 문제가 발생한다. Liu 등은 예제 모션 클립에서 빠른 접촉 변화를 나타내는 제어 그래프로 이 문제를 해결했다[22]. Han 등은 부드러운 접촉 역할을 활용하

여 불연속적인 모션 문제를 해결했다[21]. 그들의 방법은 참조 모션 추적에 중점을 두었으며, 움직이는 외부 물체와 캐릭터의 상호작용을 고려하지 않았다.

로봇 공학에서는 인간형 로봇이 축구공을 가지고 놀려는 시도가 있었다[23,24]. 휴머노이드 로봇의 복잡한 물리적 메커니즘으로 인해 축구 동작은 고정식 발차기 또는 보행 차기로 제안되었다.

이외에 캐릭터 애니메이션 분야에서도 축구 모션 합성을 위한 다양한 기술들이 제안되었다. Jain과 Liu는 사람과 축구공의 상호작용을 위한 인터랙티브 모션 편집 방법을 제안했다[25]. 이 방법은 신체부위와 접촉을 고려하여 간단한 강체 역학을 기반으로 공의 탄도 경로를 자동으로 생성했다. 공과 신체부위 사이의 접촉 타이밍과 위치는 사용자가 수정으로 지정하였다. 인간의 움직임과 탄도 기반의 공의 경로 편집에 초점을 맞춘 이 방법은 전신 역학을 고려하지 않았으며, 탄도의 특성상 직진성이 강한 움직임만을 제한적으로 표현했다. Choi 등은 End-effector 속도와 같은 운동학적 정보를 분석하여 핸드 저글링, 축구 저글링, 농구 드리블 등 2D 저글링 동작을 생성할 수 있는 데이터 기반 방법을 제안했다[26]. Ding 등은 엉덩이를 고정한 상태에서 축구공을 차는 간단한 동작을 효과적으로 생성하고 검색하기 위한 저차원 선형 피드백 전략을 제안했다[27].

III. The Proposed Scheme

1. Getting hand position from Leap motion device

본 논문에서는 립모션 디바이스를 이용하여 손의 위치를 추적한다. 내부적으로 Microsoft에서 만든 키넥트(Kinect)와 비슷한 원리인 깊이 카메라를 이용한다. 하지만, 키넥트보다 200배 높은 감도를 가지며, 약 $\frac{1}{100}$ 밀리미터의 움직임까지 감지하기 한다. 키넥트가 모니터의 전신 스캔을 위해 정면에서 좀 떨어진 곳에 위치하여 동작을 인식한다면, 립모션은 화면 아래에서 위치하여 화면과 가까운 영역에서 동작을 인식하기 때문에 좀 더 직관적인 인터페이스로 다양한 분야에서 활용되고 있다.

본 논문에서는 앞에서 설명한 립모션을 이용하여 사용자의 손 위치를 얻고, 이로부터 저글링 움직임을 계산하기 위해 손 위치의 움직임이 크게 변하는 경우 공을 손이 움직이는 벡터로 이동시킨다. 이 과정을 구현하기 위해 우선 본 논문에서는 Unity3D와 립모션 디바이스를 이용하여 손의 움직임을 계산한다.

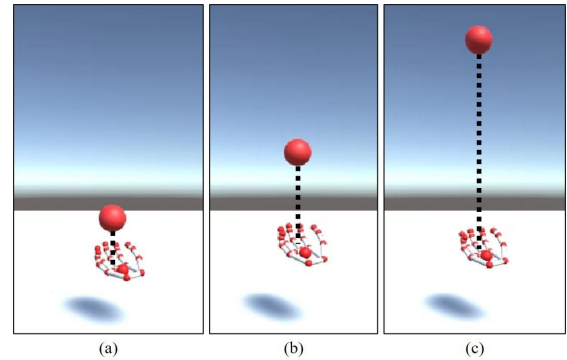


Fig. 1. Ball movement in response to hand movement ((a) small force, (b) medium strength, (c) large force).

Fig. 1은 가상환경에서 손으로 공을 제어하는 결과이다. 기본적으로 공과 손은 충돌처리를 하지 않았으며, 손의 위치 부근에 공이 위치하면 공이 손에 붙게 하였다. 그림에서 보듯이 손을 위로 튕기는 움직임을 가하면 그에 따라 공이 위로 올라가는 움직임을 보였다. 이 과정에서 손의 움직임이 빠르다면 강한 외력(F^h)으로 공을 위로 빠르게 움직이게 하였다 (수식 1 참조).

$$F = \left(m \frac{dv}{dt} \right) F^h = (ma)F^h \quad (1)$$

여기서 m 은 질량이며, v 와 a 는 속도와 가속도이다.

2. Parabolic ball movement using inverse-dynamics approach

이전 장에서 설명한 순동역학(Forward dynamics) 접근법은 외력으로 공의 움직임을 계산하는 방법이다. 이 방법을 저글링 모션에 적용하기 위해서는 한쪽 손에서 다른 손으로 이동할 때 필요한 정확한 외력을 알아야 된다. 만약, 조금이라도 외력이 다르다면 손과는 다른 위치에 공이 떨어지기 때문에 저글링 모션을 만들기 어려워진다.

저글링 모션은 포물선 형태로 움직이기 때문에 우리는 손 위치를 기준으로 거꾸로 공의 움직임을 맞춰주는 매핑 함수를 제안한다. 이를 수학적으로 공식화하기 위해 2차원 평면상에서 포물선 운동을 한다고 가정하자. 공은 평면 위를 움직인다고 가정하고, x 축에 수직인 방향으로 중력인 g 가 적용된다고 했을 때, 오일러(Euler) 적분 형태의 운동 방정식은 다음과 같다 (수식 2 참조).

$$v_z = v_{z0} \quad (2a)$$

$$v_y = v_{y0} - g\Delta t \quad (2b)$$

$$x = x_0 + v_{x0}\Delta t \quad (2c)$$

$$y = y_0 + v_{y0}\Delta t - \frac{1}{2}g\Delta t^2 \quad (2d)$$

여기서 $\{x,y\}$ 와 $\{v_x,v_y\}$ 는 각각 위치와 속도이며, Δt 는 시간간격이고, $\{v_{y0},v_{z0}\}$ 는 초기속도이다.

포물선 운동은 마치 투사체 형태로 움직이며 초기에는 속도와 중력을 받기 때문에 위 수식을 단순 조화 운동 (Simple harmonic motion)으로 변환할 수 있으며, 변환된 수식은 아래와 같다 (수식 3 참조).

$$v_{z0} = v_0\cos\theta_0\Delta t \quad (3a)$$

$$v_{y0} = v_0\sin\theta_0\Delta t - \frac{1}{2}g\Delta t^2 \quad (3b)$$

$$\Delta t = \frac{2v_0\sin\theta}{g} \quad (3c)$$

여기서 Δt 는 공이 공중에 머무는 시간을 나타낸다. 수평 최대 이동 거리를 계산하기 위해 수식 3a에 대입하면 수평 최대거리를 계산할 수 있고, 아래 수식처럼 거리 d 로부터 속도를 얻을 수 있다 (수식 4 참조).

$$d = \frac{v^2_0\sin 2\theta}{g} \quad (4a)$$

$$d \cdot g = d^2_0\sin 2\theta \quad (4b)$$

$$v^2_0 = \frac{d \cdot g}{\sin 2\theta} \quad (4c)$$

손 위치로부터 거리가 정해져 있다고 가정하여 포물선 방정식을 계산하고, 속도를 반대로 계산하면 시간당 움직이는 거리를 알 수 있다. 이것을 활용하면 단위 시간당 움직이는 포물선의 궤적을 계산할 수 있다. 수식 5a와 b는 거리로부터 계산된 공의 속도이며, 수식 5c와 d는 시간 Δt 에 따른 수평 성분과 수직 성분의 속도를 적분하여 계산한 위치이다.

$$v_x = v_0\cos\theta \quad (5a)$$

$$v_y = v_0\sin\theta - g\Delta t \quad (5b)$$

$$x = v_0\Delta t\cos\theta \quad (5c)$$

$$y = v_0\Delta t\sin\theta - \frac{1}{2}g\Delta t^2 \quad (5d)$$

앞에서 설명한 수식을 이용하여 공을 움직이면 포물선 궤적은 나타내지만, 가속도가 없기 때문에 움직임을 부자

연스럽다. 손 위치의 변화로부터 가속도를 계산하여 공의 포물선 기반 움직임에 외력을 추가시켰다 (수식 6 참조).

$$F^a = ma \quad (6a)$$

$$a = \frac{dv}{dt} = \frac{d}{dt} \frac{dp}{dt} = \frac{d^2p}{dt^2} \quad (6b)$$

공을 던지는 방향은 손의 회전 값을 계산하여 해당 방향으로 공이 투사되도록 설정했다. 공이 날아가는 도중 손의 위치에 근접되어 있을 경우, 공이 손에 붙어서 제대로 날아가지 못하는 문제가 발생한다. 이 문제를 완화하기 위해서 공을 던지는 행동을 할 경우 일정 시간동안 공이 손에 붙는 이벤트를 갖지 않도록 설정한다. 여기서의 지연 시간은 $3\Delta t$ 로 사용했다.

Fig. 2는 손의 움직임에 따라 포물선 운동을 하는 공의 움직임을 보여주는 결과이다. 립모션 디바이스로부터 얻은 손의 위치를 이용하여 움직임 벡터를 계산하고 이로부터 실시간으로 저글링 형태의 공 움직임을 모델링했다.

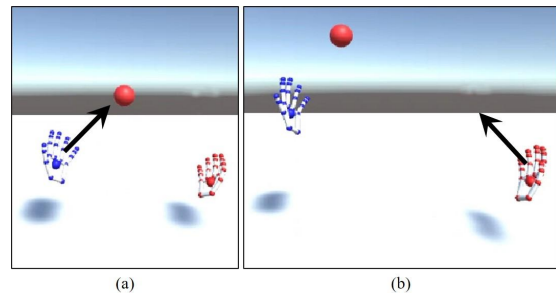


Fig. 2. Parabolic movement according to the movement of the hand (arrow : hand motion).

저글링은 양손을 사용하기 때문에 각 손과 공의 거리를 계산하여 가까운 공을 잡을 수 있도록 하였고, 앞에서 설명한 방식대로 포물선 형태로 저글링 운동이 가능하도록 설정한다. 이 과정에서 왼손과 오른손과의 거리 값, 손의 가속도 값, 기울기 값을 각각 계산하여 공을 반대로 투사시키고, 반대쪽 좌표로 공이 정확하게 날아갈 수 있도록 한다 (Fig. 3 참조).

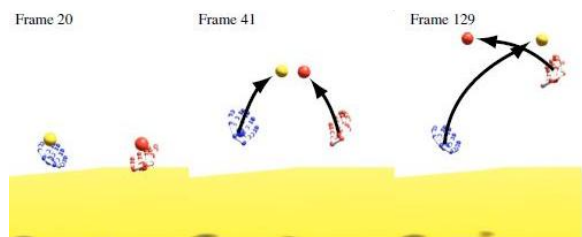


Fig. 3. Two-handed juggling motion (black arrow :ball trajectories by external force).

3. Visualizing and analyzing of arm momentum

본 논문에서는 사용자의 손목 움직임으로부터 팔 운동량을 계산하기 위해 가상의 팔목 형태를 모델링한다. 립모션 디바이스에서는 손의 위치만을 계산하기 때문에 우리는 실제로 저글링 운동을 할 때의 사람의 팔목 자세를 기준으로 가상의 팔목을 모델링하고 (Fig. 4 참조), 팔목 관절 사이의 굽힘을 이용하여 근육의 운동량을 근사한다. 실제 근육의 이완과 수축은 굽힘에 영향이 있기 때문에 본 논문에서도 이를 고려했으며, 굽힘 정도는 아래와 같이 계산하였고 (수식 7 참조), 우리는 이 값을 0~1 사이로 정규화하여 사용한다.

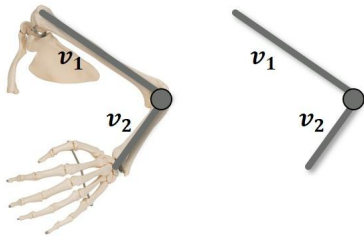


Fig. 4. Bending force between joints.

$$s_b = \cos^{-1} \left(\frac{v_1 \cdot v_2}{\|v_1\| \|v_2\|} \right) \quad (7)$$

팔목 관절로부터 근육을 운동량을 표시하기 위해 실제 팔의 전체 근육이 아닌, 팔목의 굽힘으로 인해 자주 사용하는 근육에 해당하는 부분만을 세그먼테이션(Segmentation)하여 사용한다. Fig. 5a는 실제 팔에 대한 부위별 근육을 표현한 그림이다. Fig. 5b와 같이 실제 팔과 유사한 3D 모델에서 근육의 이완과 수축에 필요한 부위별 근육만을 세그먼테이션하여 시각화에 사용한다 (Fig. 5 참조).

저글링 운동을 할 때 근육의 수축 및 이완은 크게 두 가지 움직임으로 구분된다. 공을 던지는 저글링 과정에서는 팔 근육을 접을 때와 펼 때 상황만 존재하기 때문에 본 논문에서는 두 가지 상황에 대해서만 고려를 했다 (Table 1 참조).

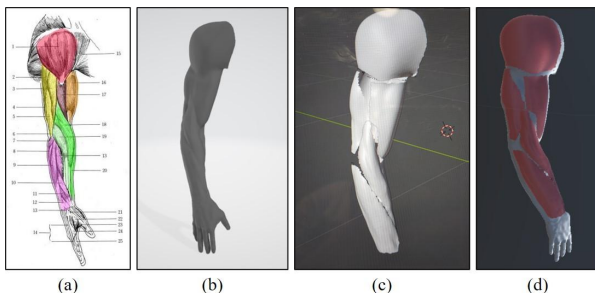


Fig. 5. Segmentation of arm muscles.

Table 1. Two cases for bending situations.

1) 팔을 접을 때 (각도가 1 이하일 때)

- 수축하는 근육 : 파란색
- 이완하는 근육 : 빨간색

2) 팔을 펼 때 (각도가 2 이상일 때)

- 수축하는 근육 : 파란색
- 이완하는 근육 : 빨간색

결과적으로 저글링을 위한 팔의 움직임을 위로 움직였을 때는 수축을 하고, 반대로 아래 방향으로 팔이 움직였을 때는 이완을 함으로써, 저글링 움직임에 대한 근육을 수축/이완을 실시간으로 관찰할 수 있다.

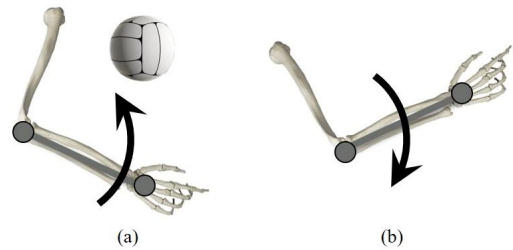


Fig. 6. Muscle contraction and relaxation calculated by bending of the joint.

Fig. 6은 관절의 움직임에 따라 근육의 수축 및 이완 상태를 결정하는 그림이다. Fig. 6a는 근육 수축으로 팔을 올리는 자세로, 저글링 움직임에서 공을 위로 튕길 때 나타나는 손의 움직임이며, 이 과정에서는 값이 0~0.5 사이의 값으로 나타난다. 근육의 이완은 저글링 동작에서 공을 던지고 난 뒤, 손을 아래로 내리면서 돌리는 과정에서 나타나는 움직임이다 (Fig. 6b 참조). 근육의 수축과 이완이 결정되면, 이 관절에 위치에 가까운 세그먼테이션된 근육에 대해서 수축/이완의 크기를 색상으로 변환하여 시각화한다.

IV. Results

본 연구의 결과들을 만들기 위해 실험한 환경은 Intel Core i7-7700K CPU, 32GB RAM, Geforce GTX 1080Ti GPU가 탑재된 컴퓨터를 이용하였다. 본 연구에서는 립모션 디바이스로부터 얻은 손의 위치를 기반으로 사용자의 동작을 분석하고, 이 동작으로부터 가상의 저글링 공을 시뮬레이션했다. 립모션 디바이스를 활용하여 손의 위치를 추적하고, 손목의 움직임에 맞는 팔과 어깨 뼈대를 추가하였고, 팔목 관절 사이의 굽힘을 이용하여 근육의 이완과 수축 상태를 시각화하였다.

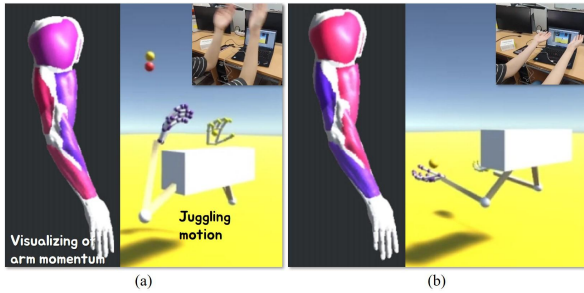


Fig. 7. Our results (contraction:■, relaxation:■).

Fig. 7은 본 논문에서 제안한 프레임워크의 결과이다. 사용자의 팔 움직임에 따라 관절의 굽힘 정도가 다르게 나타나며, 이에 따라 근육에 수축과 이완이 색상으로 시각화 되는 결과를 잘 보여준다. 우리는 립모션 디바이스를 통해 얻은 손의 위치를 이용하여 실시간으로 저글링 형태로 움직이는 가상 공을 모델링했다. 그뿐만 아니라 손의 위치와 회전 벡터를 통해 가상 관절을 추가하고, 이 관절의 굽힘을 이용하여 근육의 이완과 수축을 가시화함으로써 사용자가 현재 어떠한 근육을 사용하고 있는지 쉽게 보여준다.

Fig. 8은 왼쪽 손을 기준으로 다양한 자세를 취했을 때 그에 따른 왼쪽 근육의 변화를 시각화한 결과이다. Frame 47에서는 손을 가슴으로 밀착하여 상대적으로 안쪽으로 팔 근육이 굽힐 때의 상황을 만들어낸 결과이며, 실제와 유사한 팔의 움직임을 표현했다. (A)는 바깥쪽 근육을 나타내는 삼두근으로 팔을 굽힐 때 전형적으로 나타나는 이완형태를 잘 표현했으며, 이두근을 나타내는 (B)에서는 상대적으로 강하게 수축된 결과를 보여주고 있다.

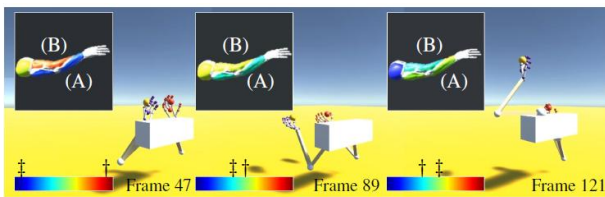


Fig. 8. Visualize muscle contraction and relaxation of left arm with our method (red : contraction, blue : relaxation, ‡ : color of (A), † : color of (B)).

이 같은 결과는 실제 팔을 굽힐 때 표현되는 근육의 변화와 같다. Frame 89에서는 손의 위치를 앞으로 이동함으로써 Frame 47보다는 좀 더 약하게 팔을 굽힌 장면이다. (A)와 (B)에서도 각각 앞에서 결과처럼 근육의 수축과 이완이 표현됐지만, 상대적으로 굽힘 정도가 약했고, 이 같은 근육의 변화가 색상으로도 잘 표현되었다. Frame 121에서는 왼쪽 손을 위로 이동하여 상대적으로 팔을 펴서, 근육이 이완된 결과를 보여주는 장면이며, Frame 47

과 89의 결과와는 반대의 성향이 나타났다 : (A)는 수축된 근육의 변화를 보였고, 상대적으로 (B)는 이완된 결과를 보여주었다. 이와 같은 결과는 실제로 팔을 굽히거나 펼 때 나타나는 특성과 동일하게 표현되었으며, 제안하는 방법이 손의 위치만을 이용하여 근육의 변화까지도 잘 캡처함을 보여주는 결과이다.

V. Conclusions

본 논문에서는 립모션 디바이스를 통해 얻은 손의 위치를 이용하여 실시간으로 저글링 형태로 움직이는 가상의 공을 모델링했다. 그 뿐만 아니라 손의 위치와 회전 벡터를 통해 가상 팔에 대한 가상 관절을 추가하고, 이 관절의 굽힘을 이용하여 팔 근육의 이완과 수축을 가시화함으로써 사용자가 현재 어떠한 근육을 사용하고 있는지 쉽게 보여준다. 본 논문에서 제안한 기법은 실시간 게임이 가능할 정도로 빠른 결과를 얻었으며, 결과를 제작하는데 60fps 이상의 성능을 보였다. 향후, 근육의 운동량을 실시간 차트로 보여주고, 어느 부분에서 잘못된 운동 하고 있고, 운동 자세를 어떻게 변경해야 하는지 가상의 트레이너 역할을 할 수 있는 방법을 추가할 계획이다.

ACKNOWLEDGEMENT

This Research was Supported by Kangnam University Research Grants. (2021).

REFERENCES

- [1] S. I. Park and J. K. Hodgins, "Data-driven modeling of skin and muscle deformation," in ACM SIGGRAPH 2008 papers, pp. 1-6, 2008.
- [2] S. Lee, M. Park, K. Lee, and J. Lee, "Scalable muscleactuated human simulation and control," ACM Transactions on Graphics (TOG), vol. 38, no. 4, pp. 1-13, 2019.
- [3] M. Romeo, C. Monteagudo, and D. S'anchez-Quir'os, "Muscle and fascia simulation with extended position based dynamics," in Computer Graphics Forum, vol. 39, no. 1, pp. 134-146, 2020.
- [4] R. Yu, H. Park, and J. Lee, "Figure skating simulation from video," in Computer graphics forum, vol. 38, no. 7, pp. 225-234, 2019.
- [5] P. Khungurn and D. Chou, "Pose estimation of anime/manga

- characters: a case for synthetic data,” in Proceedings of the 1st International Workshop on coMics ANalysis, Processing and Understanding, pp. 1–6, 2016.
- [6] J. Chemin and J. Lee, “A physics-based juggling simulation using reinforcement learning,” in Proceedings of the 11th Annual International Conference on Motion, Interaction, and Games, pp. 1–7, 2018.
- [7] L. Kumarapu and P. Mukherjee, “Animepose: Multiperson 3d pose estimation and animation,” arXiv preprint arXiv:2002.02792, 2020.
- [8] J. Won and J. Lee, “Learning body shape variation in physics-based characters,” *ACM Transactions on Graphics (TOG)*, vol. 38, no. 6, pp. 1–12, 2019.
- [9] M. Sra and C. Schmandt, “Metaspace ii: Object and full-body tracking for interaction and navigation in social vr,” arXiv preprint arXiv:1512.02922, 2015.
- [10] L. E. Potter, J. Araullo, and L. Carter, “The leap motion controller: a view on sign language,” in Proceedings of the 25th Australian computer-human interaction conference: augmentation, application, innovation, collaboration, pp. 175–178, 2013.
- [11] J.-I. Choi, S.-J. Kim, C.-H. Kim, and J. Lee, “Let’s be a virtual juggler,” *Computer Animation and Virtual Worlds*, vol. 27, no. 3-4, pp. 443–450, 2016.
- [12] D. G. Thelen, “Adjustment of muscle mechanics model parameters to simulate dynamic contractions in older adults,” *J. Biomech. Eng.*, vol. 125, no. 1, pp. 70–77, 2003.
- [13] S.-H. Lee, E. Sifakis, and D. Terzopoulos, “Comprehensive biomechanical modeling and simulation of the upper body,” *ACM Transactions on Graphics (TOG)*, vol. 28, no. 4, pp. 1–17, 2009.
- [14] W. Si, S.-H. Lee, E. Sifakis, and D. Terzopoulos, “Realistic biomechanical simulation and control of human swimming,” *ACM Transactions on Graphics (TOG)*, vol. 34, no. 1, pp. 1–15, 2014.
- [15] Y. Fan, J. Litven, and D. K. Pai, “Active volumetric musculoskeletal systems,” *ACM Transactions on Graphics (TOG)*, vol. 33, no. 4, pp. 1–9, 2014.
- [16] O. Arikan and D. A. Forsyth, “Interactive motion generation from examples,” *ACM Transactions on Graphics (TOG)*, vol. 21, no. 3, pp. 483–490, 2002.
- [17] D. Holden, J. Saito, and T. Komura, “A deep learning framework for character motion synthesis and editing,” *ACM Transactions on Graphics (TOG)*, vol. 35, no. 4, pp. 1–11, 2016.
- [18] D. Holden, T. Komura, and J. Saito, “Phase-functioned neural networks for character control,” *ACM Transactions on Graphics (TOG)*, vol. 36, no. 4, pp. 1–13, 2017.
- [19] S. Clavet, “Motion matching and the road to next-gen animation,” in *Proc. of GDC*, 2016.
- [20] M. Da Silva, Y. Abe, and J. Popović, “Simulation of human motion data using short-horizon model-predictive control,” in *Computer Graphics Forum*, vol. 27, no. 2, pp. 371–380, 2008.
- [21] D. Han, H. Eom, J. Noh, and J. S. Shin (formerly Sung Yong Shin), “Data-guided model predictive control based on smoothed contact dynamics,” *Computer Graphics Forum*, vol. 35, no. 2, pp. 533–543, 2016.
- [22] L. Liu, M. van de Panne, and K. Yin, “Guided learning of control graphs for physics-based characters,” *ACM Transactions on Graphics*, vol. 35, no. 3, 2016.
- [23] S. Barrett, K. Genter, T. Hester, M. Quinlan, and P. Stone, “Controlled kicking under uncertainty,” in *The Fifth Workshop on Humanoid Soccer Robots at Humanoids*, 2010.
- [24] T. Hester, M. Quinlan, and P. Stone, “Generalized model learning for reinforcement learning on a humanoid robot,” in *2010 IEEE International Conference on Robotics and Automation*, pp. 2369–2374, 2010.
- [25] S. Jain and C. K. Liu, “Interactive synthesis of humanobject interaction,” in *Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pp. 47–53, 2009.
- [26] J.-I. Choi, S.-J. Kang, C.-H. Kim, and J. Lee, “Virtual ball player,” *The Visual Computer*, vol. 31, no. 6-8, pp. 905–914, 2015.
- [27] K. Ding, L. Liu, M. Van de Panne, and K. Yin, “Learning reduced-order feedback policies for motion skills,” in *Proceedings of the 14th ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, 2015, pp. 83–92.

Authors



Jong-Hyun Kim received the B.A. degree in the Department of Digital Contents at Sejong University in 2008. He received M.S. and Ph.D. degrees in the Department of Computer Science and Engineering at Korea University,

in 2010 and 2016. He is an associate professor in the School of Software Application in Kangnam University. His research interests include physics-based simulation and natural phenomenon modeling and virtual production.