

# Technology Trends for Motion Synthesis and Control of 3D Character

Jong-In Choi\*

## Abstract

In this study, we study the development and control of motion of 3D character animation and discuss the development direction of technology. Character animation has been developed as a data-based method and a physics-based method. The animation generation technique based on the keyframe method has been made possible by the development of the hardware technology, and the motion capture device has been used. Various techniques for effectively editing the motion data have appeared. At the same time, animation techniques based on physics have emerged, which realistically generate the motion of the character by physically optimized numerical computation. Recently, animation techniques using machine learning have shown new possibilities for creating characters that can be controlled by the user in real time and are expected to be developed in the future.

▶ Keyword: Computer graphics, Motion capture, Data-based animation, Physics-based animation, Artificial intelligence

## I. Introduction

캐릭터 애니메이션 기술의 비약적인 발전으로 대규모 게임이나 영화에서 실사에 가까운 사실적인 캐릭터들이 대거 등장할 수 있게 되었다. 이와 같은 계기를 마련해 준 기술이 바로 모션 캡처라 할 수 있다. 모션 캡처 기술로 인해 대량의 모션 데이터를 생성하고 구축할 수 있게 되면서 모션 데이터 기반의 애니메이션 분야가 발전하게 되었다. 이는 기존의 키프레임 애니메이션을 대체하면서 비약적으로 발전하고 있다. 키프레임 애니메이션은 애니메이터의 숙련도에 비례하여 애니메이션의 품질이 결정되는 반면, 모션 캡처 애니메이션은 배우의 연기력에 비례하여 애니메이션의 품질이 결정된다. 그러나 모션 캡처 데이터를 수정하는 것은 또 다른 문제를 야기했다. 이에 캡처한 모션을 최소한의 노력으로 자연스럽게 보이도록 수정하는 것이 화두로 떠오르게 되었고 이에 대한 다양한 연구들이 등장하게 되었다. 모션 와핑, 모션 블렌딩, 모션 트랜지션 등의 모션 편집 기법들이 그것이다 [1,2,3]. 이들은 모션 캡처 데이터에서 한 프레임의 포즈를 수정 및 변경하면 이웃하는 프레임들의 포즈를 적절히 변경해 주어 자연스럽게 이어지도록 한다. 이러한 데이터 기반 방식은 자연스러운 캐릭터의 모션을 위해서 방대한 양의 모션 데이터가 필요한 문제가 있다. 때문에 모션 데이터의 양을 줄이기 위해 모션 재사용에 관한 연구가 활발히 진행되고

있다 [4,5]. 데이터 기반 방식과 더불어 로봇 공학 분야에서는 출발한 물리 기반 캐릭터도 많은 연구가 진행되었다 [6,7,8]. 물리 기반 방식은 캐릭터를 움직이는 제어를 만들고, 사용자가 입력 조건을 다양하게 지정할 수 있도록 하여 외부의 반응에 대응하는 캐릭터의 애니메이션을 생성할 수 있다. 물리 기반 방식은 사용자가 캐릭터의 디테일한 모션을 제어하는 것이 어려우므로 캐릭터의 과장된 동작을 만드는 것이 매우 까다롭다. 그러나 적은 데이터를 가지고 실제 사람이 연기하기 힘든 환경에서 애니메이션을 생성할 수 있는 것은 물리 기반 방식의 특징이라 할 수 있다. 현재 캐릭터 애니메이션은 데이터 기반 방식과 물리 기반 방식의 두 분야로 빠르게 발전하고 있으며 최근에는 두 분야를 합친 방식도 지속적으로 등장하고 있다. 물리 기반 캐릭터에 모션 데이터를 레퍼런스로 적용하는 것이 그 좋은 사례이다 [9,10,11]. 최근에는 머신러닝을 이용한 캐릭터 애니메이션이 강세를 보이고 있다 [12,13,14,15]. 알파고로 화제가 된 딥마인드에서는 강화학습을 이용하여 인간의 동작을 스스로 학습하는 물리 기반 캐릭터를 생성하였다. 이 기법으로 캐릭터 스스로 이동하고, 장애물을 뛰어 넘는 방법을 스스로 학습할 수 있다. 캐릭터의 이동 모션이 실제 사람의 모션에 비해 부자연스러워 보이지만, 모션 데이터 없이 이동 모션을 스스로

\*First Author: Jong-In Choi, Corresponding Author: Jong-In Choi

\*Jong-In Choi (gameai@ysu.ac.kr), School of Digital Culture & Contents, Youngsan University.

\*Received: 2019. 03. 06, Revised: 2019. 04. 05, Accepted: 2019. 04. 05.

Table 1. Character Animation Techniques

Division	Data-based Animation	Physics-based Animation	AI-based Animation
Characteristic	Techniques for generating animation in real time using a large amount of motion capture data. Motion blending, motion transition, motion retargeting are used.	A method of creating an animation by moving a character's bone using an actuator that moves by physical force. Controls character motion using path generator and balance controller	By using various artificial intelligence techniques including artificial neural network. It is possible to generate existing data-based and physics-based animation more realistically and effectively.
Advantages	<ul style="list-style-type: none"> <li>• Can create and control character animation intuitively more than physics-based methods</li> <li>• Since there are no exceptions, it is easy to use for moving games, movies, and animations according to the scenario.</li> </ul>	<ul style="list-style-type: none"> <li>• Appropriate animations are automatically generated in response to ambient or external forces</li> <li>• Can generate various motions with a little animation data</li> </ul>	<ul style="list-style-type: none"> <li>• Ability to create characters with less memory and faster responsiveness than existing data-based methods</li> <li>• Stable and realistic character creation compared to existing physics-based method</li> </ul>
Disadvantages	<ul style="list-style-type: none"> <li>• Large amount of motion data is required to generate realistic animations</li> <li>• Sometimes it seems unnatural in certain circumstances because physical force is not applied</li> </ul>	<ul style="list-style-type: none"> <li>• Because there are various animations generated by surrounding environment and force, occasionally control is impossible (If excessive force is applied, it can cause animation crash)</li> </ul>	<ul style="list-style-type: none"> <li>• Character motion training takes a lot of time</li> <li>• If it does not work properly, it is hard to know the exact cause, so it is very difficult to fix</li> </ul>

학습할 수 있다는 사실에 의미가 크다. 이는 레퍼런스 데이터 없이 캐릭터의 이동이라는 목적 달성만을 위해 무작위 모션 데이터를 생성하기 때문이다. 데이터 기반 방식에 신경망을 적용하여 반응성을 극대화시킨 애니메이션 기법도 등장하였고, 물리 기반 캐릭터를 강화학습으로 훈련시켜 다양한 지형 및 환경에 대응하는 캐릭터도 생성할 수 있게 되었다. 여기에 농구공 드리블을 스스로 배우는 캐릭터라든지 드리블하며 이동하는 캐릭터의 모션을 생성하는 연구도 진행되었다. 본 연구에서는 캐릭터 애니메이션을 데이터 기반, 물리 기반, 인공지능 기반 방식으로 분류하여 각각의 기술에 대해 살펴볼 것이다. Table 1은 관련 기술에 대한 특징과 장단점을 비교한 표이다.

에 마커가 숨겨져 데이터가 누락될 수 있다. Fig. 1은 광학식 모션 캡처 장비를 이용하여 모션을 촬영하는 장면이다.



Fig. 1. Optical Motion Capture

## II. Motion Capture Technique

모션 캡처는 데이터 기반 애니메이션을 구현하는 핵심 기술이다. 모션 캡처는 액터의 몸에 센서를 부착하거나 적외선을 사용하여 액터의 움직임을 디지털로 기록하는 작업을 말한다. 이 기술의 핵심은 신체의 다양한 부위에 센서를 부착한 후 센서의 위치를 통해 가상의 캐릭터가 액터와 동일한 동작으로 움직이도록 하는 것이다. 모션 캡처는 실제 사람의 모션 데이터를 숫자 데이터로 저장하고 컴퓨터로 생성된 가상 캐릭터로 모션 데이터를 전송하는 과정이다. 모션 캡처는 광학 방식, 자기 방식, 기계 방식 및 마커리스 방식으로 분류할 수 있다 [16].

### 1. Optical motion capture

광학 방식은 물체의 움직임을 파악하고 여러 카메라의 2D 영상을 포착 한 후, 3D 위치 데이터를 다시 계산하여 모션 데이터를 획득하기 위해 액터의 적절한 신체 부위에 반사 마커를 부착하는 방식이다. 고속으로 촬영할 수 있기 때문에 데이터가 거의 손실되지 않는다. 액터의 움직임을 제한하는 요소가 없으므로 모션을 자유롭게 표현할 수 있으며 섬세한 모션의 추출이 가능하다. 여러 명의 배우가 동시에 촬영되는 경우 작동 중

### 2. Mechanical motion capture

기계 방식은 기계식 장비의 부착을 통해 인체의 각 관절 부분의 움직임을 포착한다. 장비가 신체에 부착되어 있기 때문에 실시간으로 물체의 움직임을 포착하므로 비교적 깨끗한 데이터를 수집할 수 있으며 장비가 저렴하다. 실시간으로 추출된 각 관절의 회전 데이터는 매우 정확하지만 부착된 장비가 무거우며 대상의 움직임이 제한되어 있기 때문에 모션 데이터의 추출에 한계가 있다. Fig. 2는 기계식 모션 캡처 장비를 착용하여 모션을 캡처 하는 장면이다.



Fig. 2. Mechanical Motion Capture

### 3. Magnetic motion capture

자기 방식은 인체의 각 관절에 자기장을 발생시키는 센서를 부착하고, 대상물의 움직임에 따라 자기장의 변화를 측정하여 위치 데이터를 추출하는 방식이다. 광학 유형에 비해 장비의 가격 및 유지보수 비용이 적다. 케이블을 사용하기 때문에 물체의 작동에 제한이 있으며 주위의 금속 물체 등으로 인해 데이터가 손실될 수 있다. Fig. 3은 자기식 모션캡처 장비 및 수트를 착용하고 모션을 캡처하는 장면이다.

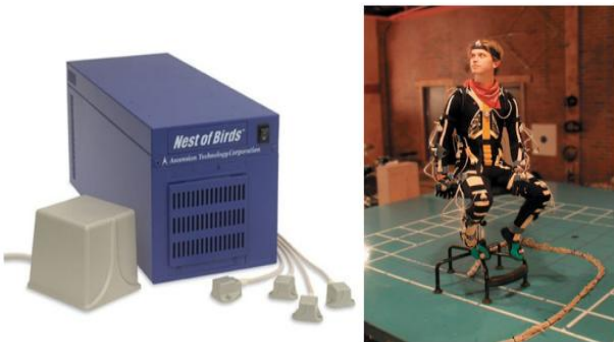


Fig. 3. Magnetic Motion Capture

### 4. Markerless motion capture

마커리스 방식은 다수의 비디오 카메라 및 이미지 처리 기술을 사용하여 마커를 사용하지 않고 실시간으로 액터의 포즈를 감지하고 추적한다. 배우의 신체에 마커 또는 센서를 부착할 필요가 없으므로 별도의 조명이 필요하지 않다. 마커가 붙어 있지 않기 때문에 애니메이션 제작뿐만 아니라 일반인에게도 모션 인터페이스에 적용할 수 있다. 그러나 모션의 정확성은 상용화하기에 부족한 경우가 많다. 마커리스 모션 캡처 시스템은 별도의 수트나 마커가 필요 없으므로 오랜 시간 동안 시스템을 설정할 필요가 없으며 마커가 가려지는 등의 오류가 없다. 특히, 기술적인 어려움이 없으며, 별도의 관리 인력이 필요 없기 때문에 현재 업계에서 사용되는 모션 캡처 기술의 가격을 크게 낮출 것으로 예상된다. Fig. 4는 마커리스 모션캡처 시스템을 이용하여 모션을 캡처하는 장면이다.

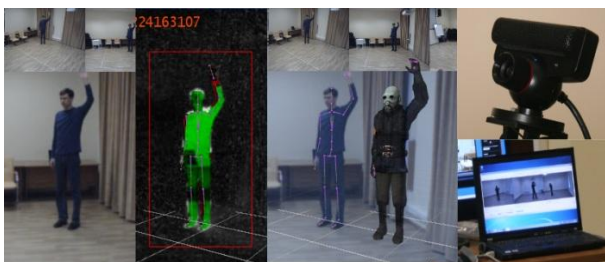


Fig. 4. Markerless Motion Capture

## III. Data-based animation

### 1. Motion editing

모션 에디팅은 사용자의 요구에 따라 모션 데이터의 일부를 편집하여 애니메이션에 적용하는 기술이다. 이러한 모션 에디팅은 1987년 Glacier가 처음 제안한 이래 계속 발전해 왔다. Glacier는 모션 캡처에서 얻은 모션 데이터의 시공간 제약 조건을 사용하여 원래 모션의 본질과 자연스러움을 보존하면서 모션을 변형하는 방법을 제시했다 [17]. 또한 캡처된 모션 데이터를 모션 리타겟팅 문제에 적용하여 원래 데이터의 자연스러움을 보존하면서 다른 캐릭터에 모션 데이터를 적용했다 [4]. 이제희 등은 역기구학과 다단계 곡선 근사법을 사용하여 모션 에디팅 및 리타겟팅의 문제를 효율적으로 해결하는 방법을 제안했다 [18]. 또한 사용자의 개입없이 모션 에디팅, 특히 모션 리타겟팅을 실시간으로 해결하는 방법이 제안되었다 [5].

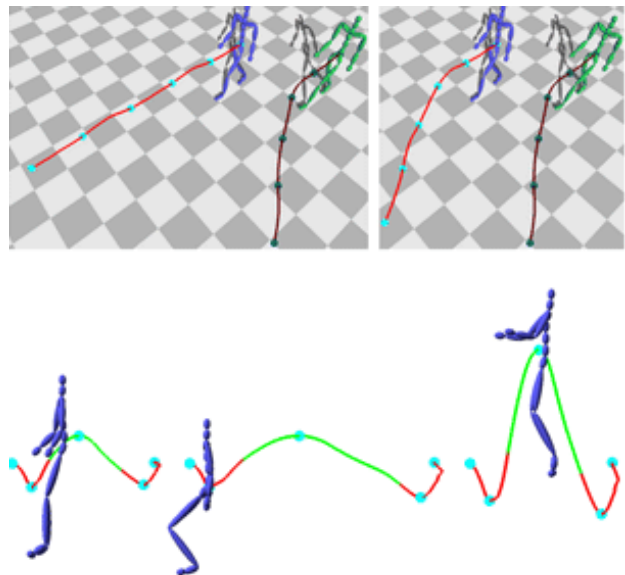


Fig. 5. Motion Path Editing

이 기술은 캐릭터의 손이나 발의 위치 및 이동 경로와 같이 사용자가 지정한 제한 조건을 만족시키면서 동작을 변형시키는 방법이다. 이 기술을 사용하면 모션 데이터를 쉽게 변형하고 애니메이션에 적용할 수 있다. 이러한 결과는 사용자의 요구에 따라 모션 데이터를 수정하는 데 사용할 수 있을뿐 아니라 모션 편집을 통해 모션이 변경된 경우에도 원래 모션의 자연스러움을 복원할 수 있다. Fig. 5는 캐릭터의 이동 경로에 따라 자연스러운 움직임을 생성하는 경로 편집 기술을 보여준다 [20]. Fig. 5는 캐릭터의 이동 경로에 따라 자연스러운 움직임을 생성하는 경로 편집 기술을 보여준다. 상단의 이미지는 이동 경로의 편집 결과이다. 좌측은 원본 이동 경로이고, 우측은 편집된 이동 경로이다. 편집된 이동 경로(붉은 색 라인)에 맞춰 캐릭터의 모션이 자동으로 보정된다. 하단의 그림은 캐릭터의 점프 모션의 경로 편집 결과이다. 좌측은 캐릭터 켈비스의 원본 이동



경로이다. 중간 이미지는 펠비스의 이동 경로를 수평으로 늘린 결과이고, 우측 이미지는 펠비스의 이동 경로를 수직으로 늘린 결과이다. 편집된 이동 경로에 맞춰 캐릭터의 모션이 자동으로 보정된다. 하늘색 점은 스플라인으로 표현된 이동 경로의 제어 점이다.

**2. Motion graph**

모션 그래프는 모션 데이터베이스에서 모션의 각 프레임의 포즈 및 연결성을 분석하여 그래프 구조에서 연결 가능한 지점을 표현하는 방법이다. 이러한 그래프 기반 모션 연결 구조 표현 방법들이 차례로 보고되었다 [19, 20]. 그래프의 형태로 모션의 프레임 또는 단위 동작의 연결을 동적으로 나타내며 그래프의 경로를 선택하고 각 노드와 에지에 저장된 동작을 연결하여 동작을 생성한다. 연속 동작은 그래프의 형태로 표현되기 때문에 데이터베이스에서 원하는 경로를 찾는 문제로 사용자가 지정한 동작을 해결할 수 있다. 그들은 평면 위에서 경로상의 움직임, 장애물 등을 고려하여 사용자 지정한 지점에 도달하는 모션, 비디오 입력과 가장 유사한 모션을 생성하기 위해 적절한 그래프 검색 방법을 제안했다. Fig. 6은 [19]의 방법에서 모션 그래프의 포즈 비교 방법을 통해 자동으로 생성된 정보를 보여준다.

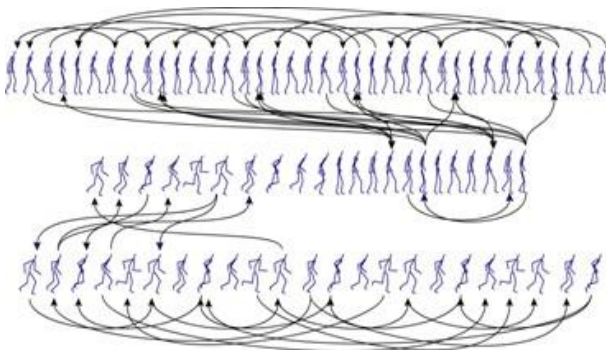


Fig. 6. Connection Information for Motion Graph

**3. Interactive motion synthesis**

다양한 캐릭터 모션 생성 기술을 적용하여 상호 작용에 중점을 둔 연구도 도입되었다. Thorne 등은 직관적인 스케치를 통해 캐릭터 애니메이션을 자동으로 생성하는 방법을 제안했다 [20]. 즉, 캐릭터의 경로뿐만 아니라 몇 가지 간단한 제스처를 입력하면 적절한 동작을 연결할 수 있다. 이 방법은 일반 사용자가 애니메이션을 간단히 만들 수 있는 수단을 제공한다. Fig. 7은 스케치 기반 접근법[20]을 사용하여 캐릭터의 모션을 효과적으로 만드는 법을 보여준다. 사용자가 펜으로 모션의 기호를 그려주면 이에 대응하여 캐릭터의 모션이 자동으로 생성된다. 또한, 모션 캡처 기술과 유사한 장비로 비디오 게임 인터페이스를 만들 것을 제안했다 [21]. 차이와 호킨스가 제안한 이 방법은 몇 개의 마크 포인트를 가진 게임 참가자가 모션을 취하면 모션 추출 정보와 유사한 특성을 지닌 데이터베이스로부터 모션을 추출하고 적절하게 연결하여 동작을 생성한다. 최근에는

모션 데이터베이스에 저장된 움직임의 일부 또는 전체를 분석하여 낮은 차원으로 투영하여 구조를 시각화하고 사용자가 저차원으로 그려주는 곡선에 해당하는 동작을 생성하는 기술이 제안되었다 [22]. 이 기법은 기본적으로 모션 데이터베이스를 차원 축소 기법으로 압축하고, 사용자의 입력이 단순할 때라도 원하는 모션을 추출하기 위해 사용하며, 모션 최적화 및 역기구학과 같은 기존 애니메이션 관련 연구에 적용되었다.

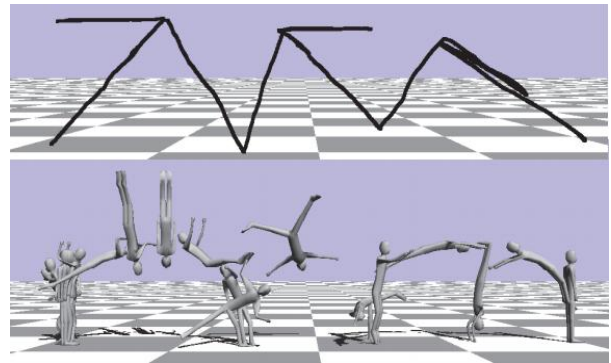


Fig. 7. Sketch-Based Motion Synthesis

**IV. Physics-based animation**

과거 전형적으로 어려운 문제로 여겨지던 물리 기반의 이족 보행 시뮬레이션은 놀라운 기술 개발로 인해 많은 문제가 해결되었다. 정상적인 보행의 경우 현실적이고 안정적인 시뮬레이션이 가능하며 보다 동적인 모션으로 연구 범위를 확장하고 있다. 캐릭터의 보행을 제어하기 위해서는 각 모션마다 컨트롤러 및 물리 시뮬레이터의 파라미터를 직접 조정해야 하지만, 장기적으로 자동화된 컨트롤러를 생성하는 것이 목표이다. 보행 컨트롤러 개발에는 두 가지 제작 방식이 있다. 첫 번째 방법은 전통적인 제어 이론과 운동학을 기반으로 한다. 두 번째 방법은 물리적 시뮬레이션이 모션 데이터를 기반으로 데이터를 근사화하는 방법이다.

**1. Control theory and path tracking**

이족 보행 시뮬레이션의 가장 중요한 부분은 원하는 모션을 생성하는 경로 생성기 및 추적기와 캐릭터의 균형을 유지하는 균형 컨트롤러이다. 캐릭터의 경로를 추적하기 위해 PD (Proportional Derivative) 서보가 주로 사용된다. 스프링 댐퍼처럼 작동하는 PD 서보는 캐릭터를 구성하고 있는 조인트 궤적의 기구학적 상태를 추적하기 위한 제어력을 계산하는 간단한 프레임 워크를 제공한다. 그러나 정밀한 추적이 필요한 경우, PD 서보는 일반적으로 불안정한 제어력을 발생시키므로 PD 서보를 사용할 때 추적의 정확도와 시뮬레이션의 효율성 사이의 적절한 값을 선택해야 한다. 시뮬레이션이 원하는 경로에서 벗어나는 경우 경로는 위치와 속도의 거리만큼 반대 방향으로 힘이 가해지는 방식으로 추적된다. PD 서보를

사용할 때, 시뮬레이션은 필요 이상으로 강화되는 경향이 있다. 이러한 현상을 피하기 위해 중력의 영향을 사전에 제거하기도 한다 [6]. 이상적으로 경로는 최소한의 힘을 사용하여 추적해야 하며, 최적 제어 이론에 기반한 경로 추적 방법을 사용해야 한다. 우선 인체의 관절 구조를 단순화하고 선형 모델로 표현할 때, 최적 경로 추적은 선형 이차 레귤레이터에 의해 설명된다 [7]. 선형 이차 레귤레이터는 간단한 구현, 낮은 계산 복잡성 및 안정성과 같은 많은 장점을 가지고 있다. 그러나 자유도가 너무 높아 선형 모델로서 인간의 골격 움직임을 표현할 때 비선형적인 특징을 갖는다. 보다 보편화된 비선형 이차 레귤레이터가 보형 시뮬레이션에서 성공적으로 사용된 경우도 있다 [8]. Fig. 8은 물리 기반 캐릭터의 방향 전환을 위한 연속 동작과 발바닥에 작용하는 힘을 시각적으로 나타낸 것이다 [23].

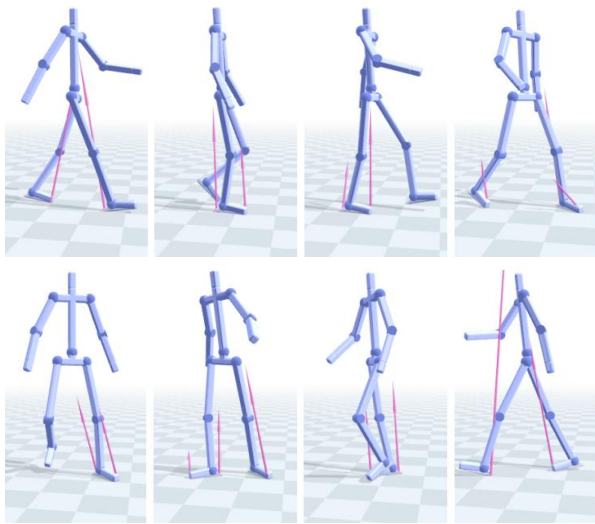


Fig. 8. Change Direction of a Character

## 2. Balance control

시뮬레이션된 이족 캐릭터가 움직이지 않고 정적 균형을 유지하는지 여부는 무게 중심이 바닥에 투영되는 지점이 바닥면에 포함되도록 보장하는 것이다. 지지면은 양쪽 접촉점을 바닥면과 함께 포함하는 볼록 다각형으로 정의된다. 그러나 동작중인 캐릭터의 동적 균형은 명확하게 정의되어 있지 않다. 이족 보행의 균형을 제어하기 위해 세 가지 접근법이 사용된다. 첫 번째 단계는 매 단계마다 발을 내딛는 위치를 번갈아 조정하여 균형을 유지하는 것이다. 예를 들어, 몸이 앞으로 기울어지면 걷는 거리를 늘려서 속도를 감소시키고, 몸이 뒤로 기울어지면 걷는 거리를 줄여서 속도를 증가시킨다. 이는 매우 간단하지만 효과적이고 안정적이며 실제 사람이 균형을 이루는 방식과 매우 유사하다. 두 번째는 캐릭터가 특정 시간 동안 자신이 원하는 경로를 따라갈 수 있다면 계속해서 균형을 유지할 수 있다는 것이다. 시뮬레이션의 매 순간마다 일정한 시간을 설정하고 이 시간 동안 최적의 추적 경로가 수치 최적화를 통해 계산된다. 일반적으로, 최적화는 비선형 이차 프로그래밍으로 공식화된다 [7]. 세 번째는 기계 학습 방법으로서 인간의 보행

데이터의 회귀 분석으로부터 평형 전략을 계산하거나 최적화된 기법과 함께 학습된 선형 평형 모델을 사용하는 것이다 [8]. Fig. 9는 외부 힘에 반응하여 좁은 공간에서 균형을 이루는 물리 기반 캐릭터의 반응 동작을 보여준다 [24].

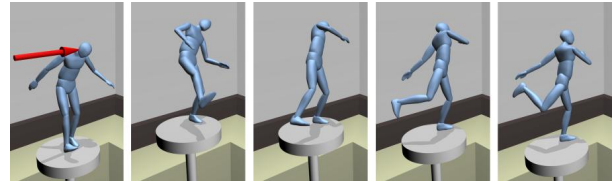


Fig. 9. Balancing a character

## V. AI-based animation

### 1. Motion synthesis using neural network

캐릭터의 모션 생성과 관련하여 Holden 등은 모션 캡처 장치에 의해 생성된 모션 데이터로부터 신경망을 배웠고, 사용자의 입력에 따라 캐릭터의 모션을 생성하는데 사용하였다. PFNN (Phase-Functioned Neural Network)은 다양한 지형에 적합한 애니메이션을 자동으로 생성해 준다 [12]. Fig. 10는 PFNN의 구조와 입력 및 출력 값을 보여준다. 보행 동작을 네 단계로 나누고 각 단계마다 적절한 네트워크 가중치를 별도로 습득한다. 위상간의 보간은 삼차 스플라인으로 수행되었다. 학습된 네트워크에 의해 생성된 스플라인은 세 개의 레이어로 구성된 완전히 연결된 네트워크의 가중치를 나타낸다. 이 완전히 연결된 네트워크는 캐릭터의 조인트 위치와 현재 시간의 이동 경로를 수신하고 다음번 조인트의 위치를 예측한다. 이런 방식으로 슬라이딩 윈도우 방법 대신 전체 애니메이션은 한 번에 하나의 프레임에 쌓아서 만들어진다. 이것은 실시간 캐릭터 조작에 보다 적합하다. 이러한 네트워크를 배우기 위해 학습 데이터가 상승 및 하강의 움직임을 배우고, 해당 지형의 높이 맵, 스타일, 각 프레임에 해당되는 도보 상태를 추가로 배운다. 생성된 모션은 아이들, 워크, 런, 점프, 크로우치 또는 그들 사이의 움직임 전환과 같은 다양한 스타일을 만들 수 있다.

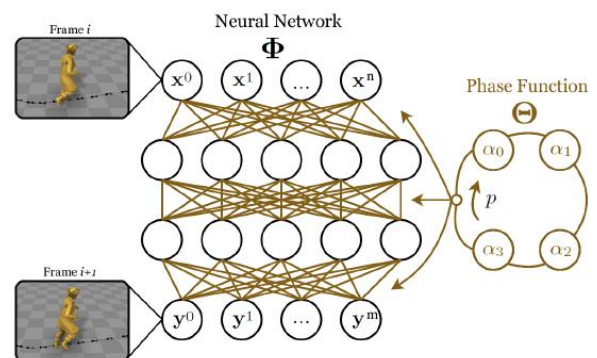


Fig. 10. Phase-Functioned Neural Network

Zhang 등은 인간의 이족 보행이 아닌 동물의 형태로 사족 보행 캐릭터의 애니메이션을 만들기 위해 PFNN을 발전시킨 MANN(Mode-Adaptive Neural Network)을 제안했다 [13]. 사족 보행 캐릭터는 27개의 관절과 81개의 자유도를 가지므로 상대적으로 복잡하다. MANN은 이전 프레임에서 위치 및 속도, 방향과 같은 정보를 수신하여 게이트웨이 네트워크 및 모션 예측 네트워크로 전송한다. 게이트웨이 네트워크는 입력 캐릭터의 상태 및 제어 변수의 서브셋을 수신하고 전문가 가중치를 출력한다. 이 가중치는 전문가 가중치와 결합되어 모션 예측 네트워크의 가중치로 사용된다. 전문가 가중치는 각 특성 운동을 별도로 학습하는 가중치이다. 모션 예측 네트워크는 캐릭터의 상태 및 제어 변수를 수신하고 다음 프레임의 캐릭터 모션을 출력한다. MANN은 목표 방향 및 속도와 같은 입력 정보와 일치하는 사족 보행 캐릭터의 애니메이션을 생성하는데 사용된다. 이전 관찰 및 사용자 입력을 수신하여 현재 프레임의 특성을 예측하여 지속적으로 변경하는 사용자의 입력에 대응할 수 있다. Fig. 11에 보이는 바와 같이 사족 보행 캐릭터가 굽은 언덕 지형뿐만 아니라 평평한 지면을 걷는 것을 학습하기 위해 다양한 데이터가 사용될 수 있다.

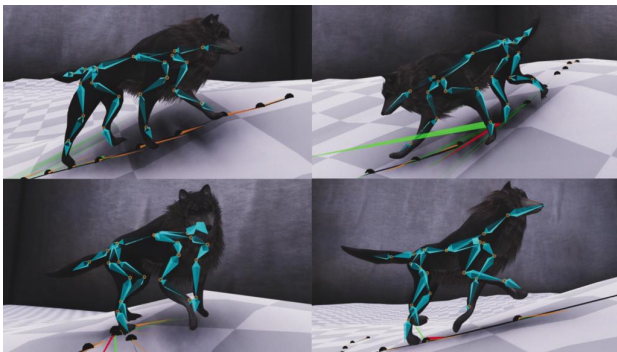


Fig. 11. A Four-Legged Character Created by MANN

**2. Motion simulation using reinforcement learning**

강화 학습은 인간이나 동물이 행동을 배우는 방식과 비교적 비슷하다. 에이전트는 주어진 환경에서 어떤 종류의 보상이 주어지는지 학습하여 소기의 목적을 달성하기 위한 행동 정책을 습득한다. 이 시행착오를 통해 접근하는 방식은 물리 계산을 통해 모션을 제작하는 물리 기반 기법에서 잘 작동한다. 펄과 페인은 이족 보행 제어기 DeepLoco와 동적 운동 제어기 DeepMimic을 출시했으며, 이 두 연구는 원하는 동작을 얻기 위해 반복적으로 관절의 힘을 변화시킴으로써 얻은 시뮬레이션 결과로부터 제어기를 향상시키는 강화 학습을 사용했다.

DeepLoco는 캐릭터 주위의 지형에 따른 이동을 계획하고 자연스러운 보행 동작을 생성하는 두 단계 프레임 워크를 제안했다 [14]. 상위 제어기는 목표 지점에 도달하기 위한 캐릭터의 경로 탐색과 경로상의 기대되는 발의 위치를 담당하며, 하위 제어기는 실제 캐릭터의 발을 움직여 보행 시뮬레이션에 필요한 제어를 담당한다. 주어진 발의 예상 위치에 따라 다른 기능

을 가진 두 제어기는 별도의 강화 학습을 사용하는 신경망으로 구성되며, 상위 제어기의 출력은 하위 제어기의 입력으로 사용되며 교차점에서 학습된다. 상위 제어기와 달리 하위 제어기는 지형에 대한 정보없이 보행 동작에 필요한 발의 위치만 받는다. 하위 컨트롤러는 현재 지원되는 발 및 움직이는 발에 상대적으로 움직이는 발의 움직임을 시뮬레이트하는 동작 정책을 학습한다. 캐릭터가 땅에 쓰러져 버리면 실제로 간주하고 보상을 0으로 설정하여 안정된 보행 제어가 가능한 제어기를 얻는다. 상위 제어기는 하위 컨트롤러의 입력이 움직이는 발의 다음 발로 위치를 출력한다. 이 때 하부 제어기의 파라미터는 고정되어 있으며 상부 제어기의 입력에 따라 원하는 시뮬레이션 결과가 얻어 졌는지를 관찰함으로써 시행착오를 통해 학습된다. 따라서 동일한 서브 제어기 내에서 다른 용도의 상위 제어기를 학습하여 사용할 수 있게 된다. Fig. 12는 해당 기술에 의해 생성된 캐릭터의 결과 장면이다. 상단의 그림은 다양한 너비의 길을 통과하는 장면이고, 하단의 그림은 볼을 드리블하며 이동하는 장면이다.

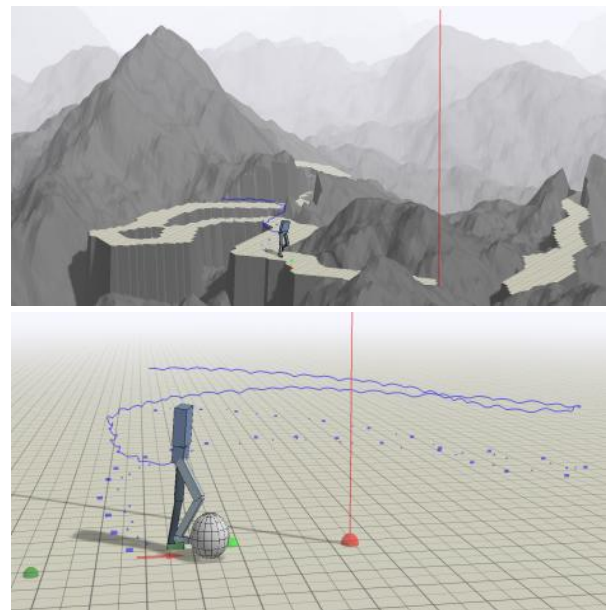


Fig. 12. Learned Locomotion Skills

DeepMimic은 강화 학습을 사용하여 데이터 기반 애니메이션 기법을 물리 시뮬레이션에 적용했다 [15]. 데이터 기반 기법은 자연스러운 동작 자체를 표현할 수 있지만 대화식 환경에서 변경된 외부 조건에 응답하지 못하는 단점이 있다. 반면, 물리 기반 기법은 외적인 요인에 의해 적절한 동작을 생성할 수 있지만, 복잡한 동작을 자연스럽게 시뮬레이션하는 것은 어렵다. Deep Mimic은 시뮬레이트된 결과가 주어진 모션 데이터와 유사한 경우 제어기가 큰 보상을 제공함으로써 학습자가 배울 수 있는 강화 학습 방법을 소개했습니다. 또한 다양한 모션을 연속적으로 처리하기 위한 다중 정책 알고리즘을 제안한다. 특정 동작을 설명하는 각 동작 데이터에 대한 개별 동작 정책을



학습한 후, 원-핫 벡터를 사용하여 특정 동작 정책을 활성화하는 방법이다. 하나의 네트워크로 원-핫 벡터를 이용하여 지정된 스킬을 제어할 수 있으며, 다른 스킬을 계속 활성화하면 해당 동작이 Fig. 13과 같이 다양한 지형에 적응하며 시뮬레이션된다.

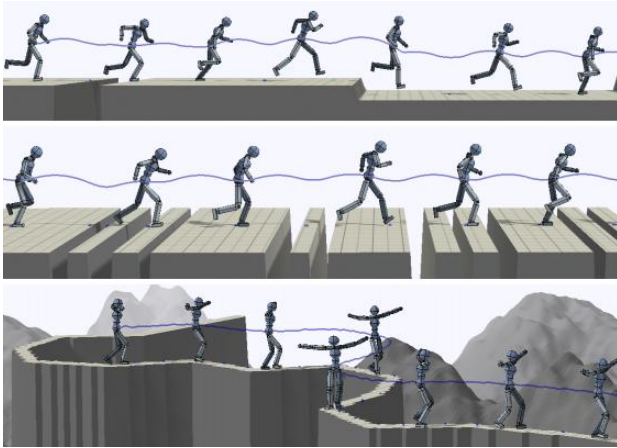


Fig. 13. Characters Traversing Terrains

## VI. Conclusions

본 연구에서는 캐릭터 애니메이션의 발전에 따른 기술의 동향에 대해 살펴보았다. 데이터 기반 애니메이션은 애니메이터의 수작업을 최소화시키고 사용자가 손쉽게 캐릭터의 애니메이션을 생성할 수 있는 방향으로 연구가 진행되고 있다. 물리 기반 캐릭터 애니메이션은 주변 환경에 반응하여 능동적으로 모션을 생성하는 방향으로 발전하고 있다. 데이터 기반 방식은 실제로 게임이나 영화에 많이 사용되고 있는 반면, 물리 기반 방식은 아직까지 상업적으로 활용되는 사례가 많지는 않다. 이는 사용자의 의도대로 모션이 생성되지 않는 경우가 많고 과장되거나 멋진 동작의 생성이 상대적으로 어렵기 때문이다. 그러나 최근 인공지능에서 화두로 떠오르고 있는 딥러닝의 발전에 힘입어 물리 기반 캐릭터의 품질이 기하급수적으로 높아지고 있으며 실시간 제어 부분도 비약적으로 발전하고 있다. 특히 농구공 드리블을 스스로 배우는 캐릭터의 경우 드리블 모션을 별도로 캡처할 필요가 없을 정도로 품질이 향상되었다. 물리 기반 캐릭터가 모션 데이터를 참고하여 이동하거나 장애물을 넘는 등의 환경에 적응하는 방식을 스스로 터득한다면 데이터 기반의 단점인 대량의 데이터 확보가 불필요해질 것이고, 이는 비약적인 비용 절감으로 이어져 상업화를 앞당길 수 있을 것이다. 또한 데이터 기반 방식의 고질적인 문제였던 느린 반응성도 놀라울 정도로 개선되었다. PFNN으로 이족 보행 캐릭터의 실시간 반응성이 크게 향상되었고, MANN을 이용하면 반응성 높은 사족 보행 캐릭터를 생성할 수 있다.

현재 데이터 기반의 애니메이션은 사용자 제어가 비교적 쉽기 때문에 게임, 영화, 애니메이션 등의 콘텐츠 분야에 활발히 사용되고 있다. 물리 기반 애니메이션은 많은 연구가 진행되고 있으나 아직까지 활용 분야는 많지 않은 편이다. 그러나 최근에 인공지능과 물리 기반 캐릭터가 접목되면서 실제 로봇의 움직임에 이를 활용하고자 시도하고 있다. 가령 어떤 물체를 옮기는 로봇 팔을 제작할 때 인공지능 기법을 사용하여 원하는 물체를 원하는 방식으로 옮기도록 훈련시킬 수 있다. 데이터 기반 애니메이션과 인공지능이 결합된 방식은 현재 여러 스포츠 게임에 적용되고 있다. 특히 PFNN과 매우 유사한 방식이 유비소프트에서 개발된 For Honor라는 콘솔 게임에서 사용되어진 사례가 있다. 캐릭터 애니메이션과 인공지능의 결합은 누구도 예상하지 못한 높은 시너지 효과를 내고 있는 분야 중 하나이다. 이제 초기 단계인 만큼 얼마나 더 발전할 수 있을지 기대를 모으고 있으며 많은 학자들이 연구에 매진하고 있는 만큼 우리의 상상을 뛰어넘는 훌륭한 결과들이 나올 것으로 기대한다.

## REFERENCES

- [1] A. Witkin and Z. Popovic, "Motion warping," in Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '95, pp. 105-108, ACM, 1995.
- [2] D. J. Wiley and J. K. Hahn, "Interpolation synthesis of articulated figure motion," IEEE Comput. Graph. Appl., vol. 17, pp. 39-45, Nov. 1997.
- [3] C. Egbert, P. K. Egbert, and B. S. Morse, "Real-time motion transition by example," in Proceedings of the 6th International Conference on Articulated Motion and Deformable Objects, AMDO '10, pp. 138-147, Springer-Verlag, 2010.
- [4] M. Gleicher, "Retargetting motion to new characters," in Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '98, pp. 33-42, ACM, 1998.
- [5] K. J. Choi and H.-S. Ko, "On-line motion retargetting," in Proceedings of the 7th Pacific Conference on Computer Graphics and Applications, PG '99, pp. 32-, IEEE Computer Society, 1999.
- [6] S. Coros, P. Beaudoin, and M. van de Panne, "Generalized biped walking control," ACM Trans. Graph., vol. 29, pp. 130:1-130:9, July 2010.
- [7] U. Muico, Y. Lee, J. Popović, and Z. Popović, "Contact-aware nonlinear control of dynamic characters," ACM Trans. Graph., vol. 28, pp. 81:1-81:9, July 2009.

- [8] K. Yin, K. Loken, and M. van de Panne, "Simbicon: Simple biped locomotion control," *ACM Trans. Graph.*, vol. 26, July 2007.
- [9] Nguyen, N., Wheatland, N., Brown, D., Parise, B., C. Karen Liu, V.Z.: Performance capture with physical interaction. In: *Proceedings of ACM SIGGRAPH/ Eurographics Symposium on Computer Animation*, pp. 189–195 (2010)
- [10] Zordan, V., Macchietto, A., Medin, J., Soriano, M., Wu, C.C., Metoyer, R., Rose, R.: Anticipation from example. In: *Proceedings of ACM Symposium on Virtual Reality Software and Technology* pp. 81–84 (2007)
- [11] Zordan, V.B., Majkowska, A., Chiu, B., Fast, M.: Dynamic response for motion capture animation. *ACM Trans. Gr.* 24(3), 697–701 (2005)
- [12] D. Holden, T. Komura, and J. Saito, "Phase- functioned neural networks for character control," *ACM Trans. Graph.*, vol. 36, pp. 42:1–42:13, July 2017.
- [13] H. Zhang, S. Starke, T. Komura, and J. Saito, "Mode-adaptive neural networks for quadruped motion control," *ACM Trans. Graph.*, vol. 37, pp. 145:1–145:11, July 2018.
- [14] X. B. Peng, G. Berseth, K. Yin, and M. Van De Panne, "Deeploco: Dynamic locomotion skills using hierarchical deep reinforcement learning," *ACM Trans. Graph.*, vol. 36, pp. 41:1–41:13, July 2017.
- [15] X. B. Peng, P. Abbeel, S. Levine, and M. van de Panne, "Deepmimic: Example-guided deep reinforcement learning of physics-based character skills," *ACM Trans. Graph.*, vol. 37, pp. 143:1–143:14, July 2018.
- [16] M. Lee, S. Park, G. Park, S. Yang, B. Lee, "Motion Capture Technology Trends," *ETRI Electronic and Telecommunications Trends*, vol. 22, no. 4, pp. 35–42, 2007. 8.
- [17] M. Gleicher, "Motion editing with spacetime constraints," in *Proceedings of the 1997 Symposium on Interactive 3D Graphics, I3D '97*, pp. 139–149, ACM, 1997.
- [18] J. Lee and S. Y. Shin, "A hierarchical approach to interactive motion editing for human-like figures," in *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '99*, pp. 39–48, ACM Press/Addison-Wesley Publishing Co., 1999.
- [19] L. Kovar, M. Gleicher, and F. Pighin, "Motion graphs," *ACM Trans. Graph.*, vol. 21, pp. 473–482, July 2002.
- [20] O. Arikan and D. A. Forsyth, "Interactive motion generation from examples," *ACM Trans. Graph.*, vol. 21, pp. 483–490, July 2002.
- [21] M. Thorne, D. Burke, and M. van de Panne, "Motion doodles: An interface for sketching character motion," *ACM Trans. Graph.*, vol. 23, pp. 424–431, Aug. 2004.
- [22] J. Chai and J. K. Hodgins, "Performance animation from low-dimensional control signals," *ACM Trans. Graph.*, vol. 24, pp. 686–696, July 2005.
- [23] U. Muico, J. Popović, and Z. Popović, "Composite control of physically simulated characters," *ACM Trans. Graph.*, vol. 30, pp. 16:1–16:11, May 2011.
- [24] A. Macchietto, V. Zordan, and C. R. Shelton, "Momentum control for balance," *ACM Trans. Graph.*, vol. 28, pp. 80:1–80:8, July 2009

### Authors



Jong-In Choi received the Ph.D. in Korea University in 2016 from the Department of Computer Science and Engineering from Korea University. He joined Nexon Korea as a lead client programmer. He also has worked at NCSOFT Korea R&D and AI

Center. Now he is a professor at the major of game contents in Youngsan University.