

# 제한된 모션 센서와 애니메이션 데이터를 이용한 캐릭터 동작 제어

배태성<sup>1,○</sup> 이은지<sup>1</sup> 김하은<sup>1</sup> 박민지<sup>2</sup> 최명걸<sup>1,\*</sup>

<sup>1</sup>가톨릭대학교 미디어기술콘텐츠학과 <sup>2</sup>티팟스튜디오(주)

{ckws5341, lej5769, kimhe045}@gmail.com minji@tpotstudio.net mgchoi@catholic.ac.kr

## Character Motion Control by Using Limited Sensors and Animation Data

Tae Sung Bae<sup>1,○</sup> Eun Ji Lee<sup>1</sup> Ha Eun Kim<sup>1</sup> Minji Park<sup>2</sup> Myung Geol Choi<sup>1,\*</sup>

<sup>1</sup>Dept. of Media Technology and Media Contents, The Catholic University of Korea <sup>2</sup>TpotStudio

### 요약

디지털 스토리텔링에 등장하는 3차원 가상 캐릭터에는 외형뿐만 아니라 자세나 동작에서도 캐릭터의 개성이 반영된 고유의 스타일이 부여된다. 그러나 사용자가 웨어러블 동작센서를 사용하여 직접 캐릭터의 신체 동작을 제어하는 경우 캐릭터 고유의 스타일이 무시될 수 있다. 본 연구에서는 가상 캐릭터를 위해 제작된 소량의 애니메이션 데이터만을 이용하는 검색 기반 캐릭터 동작 제어 기술을 사용하여 캐릭터 고유의 스타일을 유지하는 기술을 제시한다. 대량의 학습 데이터를 필요로 하는 기계학습법을 피하는 대신 소량의 애니메이션 데이터로부터 사용자의 자세와 유사한 캐릭터 자세를 직접 검색하여 사용하는 기술을 제안한다. 제시된 방법을 검증하기 위해 전문가에 의해 제작된 가상현실 게임용 캐릭터 모델과 애니메이션 데이터를 사용하여 실험하였다. 평범한 사람의 모션캡처 데이터를 사용했을 때와의 결과를 비교하여 캐릭터 스타일이 보존됨을 증명하였다. 또한 동작센서의 개수를 달리한 실험을 통해 제시된 방법의 확장성을 증명하였다.

### Abstract

A 3D virtual character playing a role in a digital story-telling has a unique style in its appearance and motion. Because the style reflects the unique personality of the character, it is very important to preserve the style and keep its consistency. However, when the character's motion is directly controlled by a user's motion who is wearing motion sensors, the unique style can be discarded. We present a novel character motion control method that uses only a small amount of animation data created only for the character to preserve the style of the character motion. Instead of machine learning approaches requiring a large amount of training data, we suggest a search-based method, which directly searches the most similar character pose from the animation data to the current user's pose. To show the usability of our method, we conducted our experiments with a character model and its animation data created by an expert designer for a virtual reality game. To prove that our method preserves well the original motion style of the character, we compared our result with the result obtained by using general human motion capture data. In addition, to show the scalability of our method, we presented experimental results with different numbers of motion sensors.

**키워드:** 캐릭터 애니메이션, 실시간 동작 제어, 가상현실

**Keywords:** Character Animation, Interactive Motion Control, Virtual Reality

## 1. 서론

게임 등의 디지털 스토리텔링에 사용되기 위해 창조된 가상 캐릭터에는 창작자가 의도한 뚜렷한 개성이 부여된다. 캐릭터의 개성은 캐릭터의 외형뿐만 아니라 자세나 동작에서도 고유한 스타

일로 나타난다. 캐릭터 동작의 스타일은 스토리텔링에 중요한 요소이므로 가능한 창작자가 만든 그대로 표현되는 것이 마땅하다. 예를 들어 버튼으로 조작하는 전통적인 비디오 게임의 경우, 사용자가 같은 상황에서 같은 점프 버튼을 누르더라도 현재 선택된

\*corresponding author: Myung Geol Choi/The Catholic University of Korea(mgchoi@catholic.ac.kr)

캐릭터에 따라 서로 다른 스타일의 동작이 플레이되어야 한다. 사용자 입력의 자유도는 제한적이지만 창작자가 의도한 캐릭터 고유의 스타일이 온전히 표현될 수 있다는 장점이 있다. 반면 최근 VR HMD 장비의 보급과 함께 늘어나고 있는 인체 동작센서를 이용하는 게임의 경우, 사용자 신체의 움직임이 그대로 복사되어 가상 캐릭터의 동작으로 나타난다. 그 결과 사용자의 동작 제어에 대한 자유도가 무한에 가깝게 확장되지만, 창작자가 의도한 캐릭터 고유의 동작 스타일은 완전히 사라지는 부작용이 발생한다.

본 논문에서는 보편적인 VR HMD 장비에서 사용되는 머리와 양손에 대한 동작 센서를 이용하여 가상 캐릭터 인체 자세를 제어함과 동시에 창작자가 디자인한 동작 스타일을 최대한 유지하는 기술을 소개한다. 머리와 양손 외의 다른 신체 부위의 움직임을 유추하기 위해서 데이터 기반 해법을 사용한다. 게임 캐릭터를 위해 특별히 제작된 동작 데이터는 그 양이 일반적인 기계학습법의 학습 데이터로 사용하기에 턱없이 부족하다. 본 연구에서는 동작 데이터의 양이 제한적이라는 점을 역이용하여 기계학습 대신 빠른 인체 자세 검색 기법을 사용하여 문제를 해결한다. 그 결과 데이터 회귀 (Regression) 등의 중간 처리 과정 없이 창작자가 제작한 동작 데이터를 원본 그대로 사용함으로써 캐릭터 고유의 스타일을 최대한 유지할 수 있게 된다. 하지만 검색 알고리즘의 사용은 매 프레임마다 검색되는 캐릭터 자세 변화의 연속성을 보장하지 못한다는 단점이 있다. 이를 해결하기 위해 검색되는 캐릭터 자세에 대해 보정하는 단계를 거친다.

본 논문에서 제안하는 방법의 실용성을 증명하기 위해 가상 현실 게임을 위해 제작된 캐릭터 동작 데이터를 사용하여 실험을 진행한다. 또한 일반적인 사람의 동작을 캡처한 모션캡처 데이터를 사용했을 때와 결과를 비교하여 스타일이 유지됨을 검증한다. 마지막으로 사용자의 허리와 양발에 동작 센서를 추가로 장착하는 실험을 통해 본 연구에서 제시하는 방법의 확장성을 검증한다.

## 2. 관련 연구

가상 캐릭터 동작 제어 문제에서는 제어의 정확성과 인체 움직임의 사실성을 동시에 고려해야 한다. 보다 적은 계산량으로 사실적인 인체 동작의 표현을 담보하기 위해 사전에 준비된 대량의 모션캡처 데이터를 사용하는 기술들이 연구되어 왔다. 데이터 기반 기술의 기본적인 접근 방법은 캐릭터 동작에 대한 새로운 지시 사항 또는 제약 조건이 주어졌을 때, 원본 모션 데이터를 조각내고 재배치하여 그 조건에 부합하는 새로운 동작을 생성하여 보여주는 것이다. 이때 원본 데이터를 작은 단위로 조각내는 기술 [1, 2, 3], 빠르게 검색하는 기술 [4, 5, 6, 7], 조건에 맞게 데이터를 변형 [8, 9, 10, 11]하는 기술 등 다양한 모션 데이터 처리 기술이 복합적으로 사용된다. 이러한 세부 과정들을 종합적으로 그리고 효율적으로 해결하기 위한 방안으로 그래프 형태의 자료구조를 사용하는 기술이 널리 응용되고 있다 [12, 13, 14]. 본 연구에서 제안하는 방법 역시 동작 데이터의 검색과 변형 기술을 사용한다. 하지만 실제 사람의 동작을 캡처하여 얻는 모션캡처

데이터를 사용하지 않고 전문가에 의해 창작된 캐릭터 모델과 제한된 양의 애니메이션 데이터를 사용하는 것을 목표로 한다. 또한 비디오 게임 등에서 요구되는 높은 수준의 인터랙션 반응 속도를 보장하기 위해 그래프 구조에 의존하는 대신 매 프레임마다 사용자 입력을 반영한 자세를 검색하고 보정하는 방법을 제시한다.

인체 동작은 자유도가 높기 때문에 제어를 위한 입력 방법을 정의하는 것이 쉽지 않다. 가장 단순한 방법은 전통적인 비디오 게임에서와 같이 특정 동작을 지시하는 버튼 (방향키, 공격키 등)을 사용하는 것이다 [15, 16]. 사용자 입장에서 쉽고 단순하지만 동작 표현의 자유도가 낮다는 단점이 있다. 사용자가 가능하면 직관적인 방법으로 높은 자유도의 자세 및 동작 제어 신호를 입력하도록 돕기 위해 스케치를 이용하는 방법 [17], 비디오 입력을 이용하는 방법 [18, 19], 인체형 기계 장치를 이용하는 방법 등 [20, 21] 이 제안되었다. 각각의 방법은 장단점이 있고 서로 다른 응용 소프트웨어에서 사용될 수 있다.

한편, 가상현실 게임의 경우에는 웨어러블 동작 센서를 이용한 방법이 널리 사용되고 있다. 동작 센서란 인체에 착용되어 해당 부분에 대한 3차원 공간상의 위치와 방향에 대한 정보를 지속적으로 측정하는 장치이다. 일반적으로 사용자 몸에 부착하는 동작센서의 수는 인체 자세에 대한 모든 정보를 얻기에 부족하다. 예를 들어 가상현실에서 주로 사용되는 동작센서는 주로 3개 (머리와 양손에 장착)이며, 경우에 따라 23개 (양발, 허리 등에 장착)의 센서가 추가되기도 한다. 이렇게 제한된 정보로부터 전체 인체 자세를 유추하기 위해서 기계학습에 기초한 기법들이 주로 연구되어 왔다 [22, 23, 24]. 기계학습은 대량의 학습 데이터를 필요로 한다. 하지만 본 연구에서는 전문가가 창작한 가상 캐릭터 동작의 제어를 목표로 하기 때문에 애니메이션 데이터의 양이 기계학습을 적용하기에 충분하지 못한 상황을 가정한다. 본 연구에서는 제한된 데이터를 가능한 빠르게 검색하여 사용하는 것을 목표로 한다.

인체 동작 데이터 처리에서 동작 데이터에 나타나는 고유의 스타일을 다루는 문제도 흥미로운 연구 주제로 여겨지고 있다. 동작 데이터의 스타일을 정확하게 정의하기는 쉽지 않지만 일반적으로 특정 동작을 설명할 때 부사나 형용사 또는 감정으로 묘사되는 부분을 스타일로 정의하고 동사로 표현되는 부분을 순수 동작으로 정의 한다 [25, 26, 27, 28]. 예를 들어 '쑹쑹한 걷기'라는 동작 데이터가 있다면 '걷기'에 해당하는 움직임 외에 '쑹쑹함'으로 묘사된 부분을 스타일이라고 한다. 하나의 동작 데이터에서 발견되는 스타일을 추출하여 다른 동작 데이터에 적용하는 기술을 스타일 전환 (style transferring)이라고 하며 이에 대한 다양한 연구가 진행되어 왔다 [29, 30, 31, 32]. 본 연구에서 다루는 문제는 사용자의 평범한 동작에서 얻은 신호를 이용하여 캐릭터 동작을 제어함과 동시에 캐릭터 고유의 동작 스타일이 표현되도록 하는 것이므로 스타일 전환 문제로 볼 수 있다. 기존의 동작 데이터의 스타일 전환 기법은 대부분 대량의 학습 데이터를 필요로 한다. 또한 학습 결과로 얻어진 회귀 모델 [33], 뉴럴넷 모델 [34, 35] 등은 원본 데이터의 여러 부분을 서로 혼합하거나 변형하여 하나의

새로운 동작을 생성하기 때문에 창작자가 표현한 동작의 정교한 디테일이 보존되지 못할 가능성이 높다. 본 연구에서 제안한 검색에 기초한 동작 제어 방법은 원본 데이터를 그대로 사용하고 정확하고 부드러운 제어에 필요한 최소한의 변형만 적용하므로 창작자가 표현한 디테일이 최대한 보존될 수 있다.

본 연구에서 제안하는 방법은 매 프레임마다 다음의 과정을 반복한다. 먼저 제한된 개수의 동작센서로부터 사용자의 현재 자세 정보를 얻는다. 다음 캐릭터를 위해 제작된 동작 데이터를 대상으로 현재 사용자 자세와 가장 유사한 자세를 검색한다. 이때 전역 위치와 방향에 대한 정합행렬 계산을 최대한 빠르게 수행하기 위해 [6]에서 제안한 방법을 사용한다. 마지막으로 검색된 자세를 보정하여 캐릭터 자세 변화의 연속성을 유지한다. 3 절에서는 검색의 효율성을 높이기 위해 동작 데이터를 전처리하는 과정을 설명한다. 4 절에서는 빠른 검색을 위한 구체적인 검색 알고리즘을 설명한다. 5 절에서는 검색된 자세를 후처리하는 과정을 설명한다.

### 3. 동작 데이터 전처리

실시간 자세 검색의 효율성을 최대화하기 위해서 캐릭터 동작 데이터에서의 캐릭터 관절 구조를 단순화하고 검색 대상이 되는 프레임 수를 제한하기 위한 사전 샘플링 과정을 수행한다.

#### 3.1 캐릭터 자세 표현의 단순화

게임 등에 사용되는 가상 캐릭터의 관절 구조는 일반 모션캡처 데이터에서 사용되는 관절 구조보다 복잡한 경우가 많다. 예를 들어 캐릭터 머리카락의 움직임이나 옷의 펄럭임 등을 표현하기 위해 뼈대를 추가하는 경우가 있다. 경우에 따라서는 캐릭터가 사용 중인 도구나 무기 등에 뼈대를 추가하는 경우도 있다. 본 연구에서는 캐릭터의 신체 자세만을 제어 대상으로 하기 때문에 신체가 아닌 부분에 대한 캐릭터 관절 구조는 계산에서 제외하였다.

일반적으로 동작 데이터는 시간에 따른 자세의 변화로 표현되므로  $\{P_1, P_2, \dots, P_n\}$ 와 같은 집합으로 표현된다.  $n$ 은 전체 프레임의 수를 나타낸다. 하나의 자세는 다시 3차원 위치와 방향값의 집합  $P_i = \{p_0, q_0, q_1, q_2, \dots, q_m\}$ 로 표현된다.  $p_0 \in \mathbb{R}^3$ 와  $q_0 \in SO(3)$ 는 각각 해당 자세의 전역 위치와 방향값이고,  $q_i \in SO(3) (i > 0)$ 는 각 관절의 3차원 방향값이다.  $m$ 는 관절 구조에 있는 총 관절의 개수다.

검색 단계 (4 절 참고)에서는 관절의 방향값은 사용하지 않고 3차원 위치 값만을 사용한다. 동작센서에서도 방향에 대한 정보를 쉽게 취득할 수 있기 때문에 검색에 방향값을 고려하는 것은 어려운 문제가 아니지만, 본 연구의 실험을 통해 경험적으로 발견한 사실은 방향값까지 고려했을 때의 검색 결과와 오히려 좋지 못했다는 것이다. 그 이유는 게임 캐릭터의 특성상 캐릭터 고유의 자세가 일반 사용자가 묘사하기 어려운 디테일을 포함하고 있는 경우가 많기 때문이다. 예를 들어 서로 다른 캐릭터는 일반적인

서있는 자세에서 각자의 스타일에 따라 손목의 방향이 완전히 다를 수 있다. 이러한 디테일은 사용자가 무시할 수 있도록 하기 위하여 캐릭터 관절의 방향값은 검색 단계에서 무시한다.

검색에서 위치값만 사용하므로 모든 동작 데이터의 관절에 대해 미리 정기구학 (forward kinematics) 계산을 수행하여 3차원 좌표값을 계산한다. 또한 검색 단계에서 각 관절과 사용자가 사용하는 동작센서를 1대1 매칭하여 비교하므로 그에 해당하는 관절에 대한 위치 값만 남겨둔다. 결과적으로 단순화된 자세  $P_i$ 의 형태는  $\{p_1, p_2, \dots, p_k | p_i \in \mathbb{R}^3\}$ 로 표현된다.  $p_i$ 는 동작센서와 대응되는 각 관절의 3차원 위치값이고  $k$ 는 동작센서의 개수이다.

#### 3.2 데이터 샘플링

본 연구에서 사용하는 검색 알고리즘은 선형검색이다. 동작 데이터의 프레임 간격은 매우 짧기 때문에 시간적으로 인근한 프레임들은 서로 매우 유사한 자세를 나타낼 것이다. 거의 비슷한 자세들은 검색 대상에서 제외하면 검색의 효율을 높일 수 있다. 따라서 전체 동작 데이터를 샘플링하여 선택된 프레임만을 검색 대상 집합으로 지정한다.

본 연구에서는 동작 데이터의 특성에 따라 두 가지 샘플링 기법을 사용하였다. 먼저 창작자에 의해 만들어진 캐릭터 동작 데이터는 양이 제한적이기 때문에 사람이 직접 전체 프레임을 관찰하여 중요 프레임 선택하는 것이 어렵지 않다. 따라서 반드시 검색 될 필요가 있는 자세를 수동으로 선택하고 나머지 부분에 대해서는 일정한 시간 간격으로 샘플링한다. 반면, 비교적 양이 많은 일반 모션캡처 데이터의 경우에는 [6]에서 제안된 불규칙 키프레임 선택 기법을 사용하여 자동으로 샘플링한다. 이 방법은 모션 캡처 데이터의 속도 변화를 분석한 다음 속도 그래프가 지역적으로 최소값을 나타내는 순간의 자세를 선택하는 것으로 여러 논문에서 유용성이 검증되었다 [7, 36, 17].

### 4. 자세 검색

3 절에서 샘플링한 프레임에 해당하는 모든 자세를 대상으로 선형검색을 수행한다. 현재 사용자가 장착하고 있는 동작센서 위치의 집합을  $S = \{s_1, s_2, \dots, s_k | s_i \in \mathbb{R}^3\}$ 라고 할 때 최적의 캐릭터 자세  $P_o$ 는 아래와 같이 검색된다.

$$o = \underset{1 \in \Omega}{\operatorname{argmin}} d(S, P_i) \quad (1)$$

$\Omega$ 는 샘플링된 프레임 번호의 집합이다.  $P_i$ 와  $S$  사이의 차이를 정의 하는 함수  $d(S, P_i)$ 는 아래와 같이 정의된다.

$$d(S, P_i) = \frac{1}{k} \sum_{j=1}^k \|\mathbf{R}_i p_j - \mathbf{D}_j \mathbf{T} s_j\|^2$$

$\mathbf{R}_i$ 은  $P_i$ 와  $S$ 를 전역 공간에서 정합하는 변환행렬이다.  $\mathbf{D}$ 는 각 동작센서와 대응되는 캐릭터 관절사이의 차이를 보정하는 변환



행렬이고  $\mathbf{T}$ 는 캐릭터 신체와 사용자 신체 크기 차이를 보정하는 변환행렬이다.

#### 4.1 센서 보정행렬 $\mathbf{D}_j$ 와 $\mathbf{T}$ 의 계산

변환행렬  $\mathbf{D}_j$ 와  $\mathbf{T}$ 는 사용자 신체와 캐릭터 신체의 차이를 보정하는 변환행렬이므로 사용자가 처음 동작센서를 착용한 직후 한 번만 계산하면 된다. 사용자는 T-자세 (양팔을 수평으로 벌리고 다리를 모으고 서는 자세) 상태로 있을 때의 동작센서 위치를 측정하고 캐릭터가 T-자세일 때의 관절의 위치와 비교한다.

$\mathbf{T}$ 는 스케일 변환행렬이다. 사용자가 양손에 동작센서를 장착하고 있는 경우 사용자와 캐릭터의 양손 사이의 거리차의 비율을 이용하여 3차원 직교좌표계의 3축에 대해 균일하게 스케일링한다. 만일 사용자가 양손 센서에 더해 양발과 머리에 대한 센서를 착용하고 있다면, 수직 축에 대해서는 사용자와 캐릭터의 발에서부터 머리 사이의 거리차의 비율 이용하여 스케일링한다.

$\mathbf{D}_j$ 는 위치와 방향에 대한 변환행렬이다. 캐릭터와 사용자가 모두 T-자세이고 동작센서 위치에 스케일 변환  $\mathbf{T}$ 가 적용된 후에  $j$ 번째 캐릭터 관절과 동작센서 사이의 위치와 방향에 대한 차이를 나타낸다.

#### 4.2 정합행렬 $\mathbf{R}$ 의 계산

만일 캐릭터 모델과 사용자가 같은 3차원 좌표계에 있다고 가정하고 두 자세의 전역 위치와 방향을 정확히 알고 있다면 두 자세를 서로 겹치고 정합하는 일은 간단한 계산으로 처리될 수 있다. 하지만 본 연구에서는 사용자 신체에 부착되는 동작센서가 제한적이라고 가정한다. 최소의 경우 머리와 양손에 하나씩, 총 세 개의 동작센서를 착용한다고 가정한다. 이 경우 전체 신체에 대한 정확한 위치와 방향을 정의하기 어렵다. 따라서 아래와 같이 현재 캐릭터 관절의 위치와 동작센서의 위치의 오차를 최소화하는 방법으로 정합행렬을 정의한다.

$$\mathbf{R}_i = \underset{\mathbf{R}}{\operatorname{argmin}} \sum_{j=1}^k \|\mathbf{R}p_j - \mathbf{D}_j\mathbf{T}s_j\|^2 \quad (2)$$

정합행렬  $\mathbf{R}_i$ 는  $\mathbf{D}_j$ 나  $\mathbf{T}$ 와는 달리 검색 대상이되는 각각의 캐릭터 자세  $P_i$ 마다 새롭게 계산되어야한다. 따라서 가능한 계산이 단순해야한다.

변수의 수를 줄이기 위해 먼저 정합행렬  $\mathbf{R}$ 를 다음과 같이 단순화 한다. 정합에서 신체 스케일의 변형화는 금지하고 신체의 회전과 이동 변환만을 허용한다. 또한 자세가 수직축을 기준으로 한 방향으로 기울어지는 것을 금지하기 위하여 수직축 ( $y$ -축) 회전만을 허용한다. 그리고 캐릭터가 하늘에 떠 있거나 땅 속으로 들어가는 것을 금지하여 수직 축을 따라 이동하는 것은 제한한다. 따라서 정합 행렬  $\mathbf{R}$ 은 다음과 같은 호모지니어스 좌표계로 나타낼 수 있다.

$$\mathbf{R} = \begin{bmatrix} \cos(\theta) & 0 & -\sin(\theta) & t_x \\ 0 & 1 & 0 & 0 \\ \sin(\theta) & 0 & \cos(\theta) & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

빠른 계산을 위해  $\mathbf{R}$ 의 비선형적인 요소인  $\cos$ 과  $\sin$  함수를  $\alpha$ 와  $\beta$ 로 치환한다.

$$\mathbf{R}' = \begin{bmatrix} \alpha & 0 & -\beta & t_x \\ 0 & 1 & 0 & 0 \\ \beta & 0 & \alpha & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{s.t.} \quad \alpha^2 + \beta^2 = 1 \quad (3)$$

식 (3)은  $\mathbf{R}'$ 의 변환이 강성 (Rigidity)을 유지하게 하는 제약 조건이다. 식 (2)는 아래와 같이 4개의 변수  $\alpha, \beta, t_x, t_z$ 에 대한 최적화식으로 재정리할 수 있다.

$$\underset{\{\alpha, \beta, t_x, t_z\}}{\operatorname{argmin}} \left\| \mathbf{A} \begin{bmatrix} \alpha & \beta & t_x & t_z \end{bmatrix}^T - \mathbf{b} \right\|^2$$

위의 최적화는 선형 문제이므로 최소제곱해를 구하는 방식으로 쉽게 계산된다. 하지만 강성 조건인 식 (3)이 고려되지 않았으므로 이를 보정하기 위해서 아래와 같이  $\alpha$ 와  $\beta$ 값을 수정한다.

$$\alpha' = \frac{\alpha}{\sqrt{\alpha^2 + \beta^2}} \quad \beta' = \frac{\beta}{\sqrt{\alpha^2 + \beta^2}}$$

$\alpha$ 와  $\beta$ 값을 최적화 후에 보정하는 것은 결과 값에 오차를 발생시킬 가능성이 있지만, [6]의 연구에 의하면 검색을 위한 용도로 사용하기에 문제가 되지 않는 수준이다.

### 5. 검색 결과 후처리

최적의 캐릭터 자세가 결정되면 검색 과정에서 계산되는 정합행렬  $\mathbf{R}$ 을 사용하여 쉽게 가상환경 내의 적절한 위치와 방향에 캐릭터를 배치할 수 있다. 배치 후에는 검색을 위해 단순화된 자세 데이터 대신 모든 관절의 각도로 표현되는 원본 캐릭터 자세 데이터를 사용하여 후보정 처리한다 (3.1 절 참조).

검색된 캐릭터 자세는 현재 사용자 자세와 정확히 일치하지 않지만 캐릭터 고유의 스타일을 유지하는 것이 목적이므로 완전히 일치시킬 필요는 없다. 다만 캐릭터와 가상환경 내의 다른 사물 사이의 상호작용에 중요한 영향을 미치는 손과 발의 위치는 사용자가 의도한 위치와 동일하게 유지하는 것이 동작 제어에 도움이 될 가능성이 크다. 또한 검색에서 고려되지 않는 관절 방향에 대한 정보도 현재 사용자가 착용중인 동작센서와 일치되도록 변형

할 수 있다. 본 연구에서 수행한 모든 실험에서는 캐릭터의 손과 발의 위치와 방향에 대해 각각 대응하는 동작센서가 있는 경우 역기구학 (inverse kinematics) 기법을 이용하여 센서 값과 동일하게 보정하였다. 캐릭터 머리의 경우에는 방향 값만 동작 센서와 동일하게 변형하였다. 역기구학 방법은 팔, 다리, 목 부분에만 제한적으로 분석적 계산법 [8]을 적용하여 불필요한 자세 변화를 최소화하였다.

매 프레임마다 독립적으로 검색된 자세들은 변화의 연속성을 보장하지 않는다. 캐릭터 자세의 갑작스러운 변화는 동작의 사실성에 부정적인 영향을 미친다. 이를 보완하기 위해 아래와 같은 식으로 프레임당 캐릭터 자세 변화의 비율을 제한하였다.

$$P'_t = \omega P_o \oplus (1 - \omega) P_{t-1}$$

위 식은 현재 검색된 자세  $P_o$ 와 이전 프레임에서 사용된 자세  $P_{t-1}$ 를 혼합하는 식으로  $\oplus$ 는 캐릭터 자세에서 위치값의 경우에는 선형 보간법을 하고 방향값의 경우 구면 보간법을 하는 연산이라 정의한다. 본 연구에서 수행한 모든 실험에서는 프레임 타임이 0.166초였으며  $\omega$  값을 0.05로 고정하여 사용하였다. 즉, 캐릭터는 자세의 변화를 0.166초 당 5%로 제한하였다.

## 6. 실험 결과

실험에 사용된 동작 데이터는 일반 모션캡처 데이터와 전문가가 특정 캐릭터를 위해 제작한 애니메이션 데이터로 나뉘어진다. 모션캡처 데이터는 자유롭게 춤추는 동작 (약 3분), 골프 동작 (약 1분), 걷기-뛰기-앉기 등의 기본 동작 (약 3분), 태권도 동작 (약 5분), 마지막으로 기타 여러 동작 (약 3분)을 포함하고 있다. 전문가가 제작한 애니메이션 데이터는 티파트스튜디오(주)에서 제작한 캐릭터와 가상현실 게임에 적합한 점프하기, 앉기, 공격하기 등의 동작 (총 약 1분)이 포함되어있다.

### 6.1 모션캡처 데이터와 캐릭터 애니메이션 데이터 비교

첫 번째 실험에서는 무작위로 수집된 모션캡처 데이터를 사용하는 경우와 전문가에 의해서 제작된 소량의 캐릭터 애니메이션 데이터를 사용하는 경우의 결과를 비교하였다. 또한 일반적인 VR HDM 장비를 사용하는 경우를 가정하여 사용자가 머리와 양손에만 모션 센서를 장착하고 실험하였다.

그림 1은 유사한 사용자 입력값에 대한 결과 차이를 비교한 것이다. 첫 번째 열은 각기 다른 사용자 입력 자세를 나타낸다. 첫 번째 행은 왼손을 들었을 때, 두 번째 행은 앉았을 때, 세 번째 행은 고개를 뒤로 젖히는 동작을 실험하고 있다. 두 번째 열은 일반적인 모션캡처 데이터를 사용했을 때의 결과를 나타낸 것이고, 세 번째 열은 캐릭터 애니메이션 데이터를 사용했을 때의 결과를

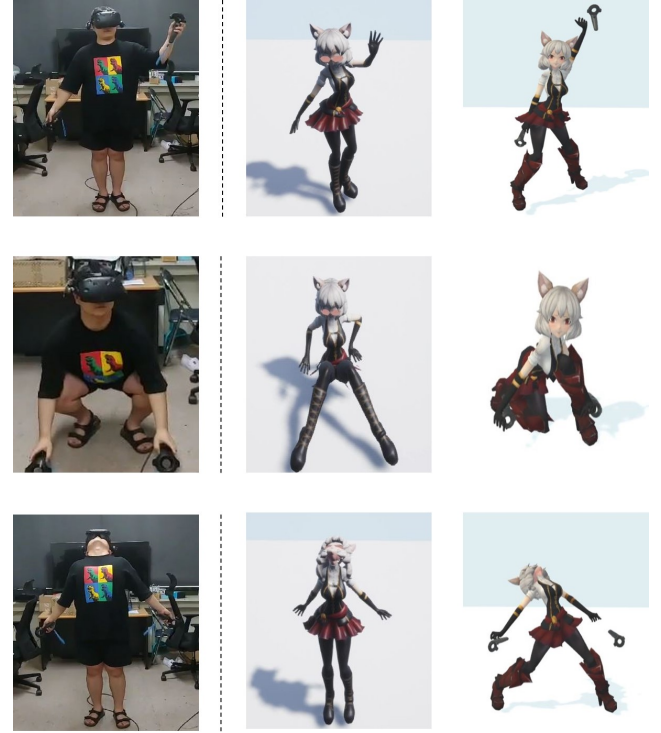


Figure 1. Comparison results of using motion capture data (middle) and animation data created by expert animators (right).

나타낸 것이다.

실험 결과 모션캡처 데이터를 사용하였을 때에는 캐릭터가 평범한 사람과 같은 동작을 하는 것을 볼 수 있었다. 또한 무작위로 수집된 모션캡처 데이터의 다양성으로 인해 동작센서가 부재한 캐릭터 다리의 움직임의 변화는 예측하기 쉽지 않았다. 반면 캐릭터 애니메이션 데이터를 사용한 경우 캐릭터 자체의 스타일이 명확하게 표현되는 것을 확인 할 수 있다. 캐릭터 다리의 움직임도 창작자에 의해 캐릭터에게 부여된 자세가 나타남을 볼 수 있다.

### 6.2 동작센서 추가에 대한 결과 비교

두 번째 실험에서는 추가 동작센서를 사용하는 경우의 결과 비교를 통해 본 연구에서 제시하는 방법의 확장성을 검증하였다. 이 실험에서는 일반 모션캡처 데이터를 사용하였으며 총 3개의 센서 (머리와 양손)를 사용하는 경우와 총 6개의 센서 (머리, 양손, 양발, 허리)를 사용하는 경우의 결과를 비교하였다.

그림 2의 첫 번째 행은 3개의 동작센서를 사용했을 때의 결과를 나타내고 두 번째 행은 6개의 동작센서를 사용했을 때의 결과를 나타낸다. 동작센서의 개수가 늘어 날수록 사용자와 자세와 보다 유사한 캐릭터 자세가 표현되었음을 확인하였다.

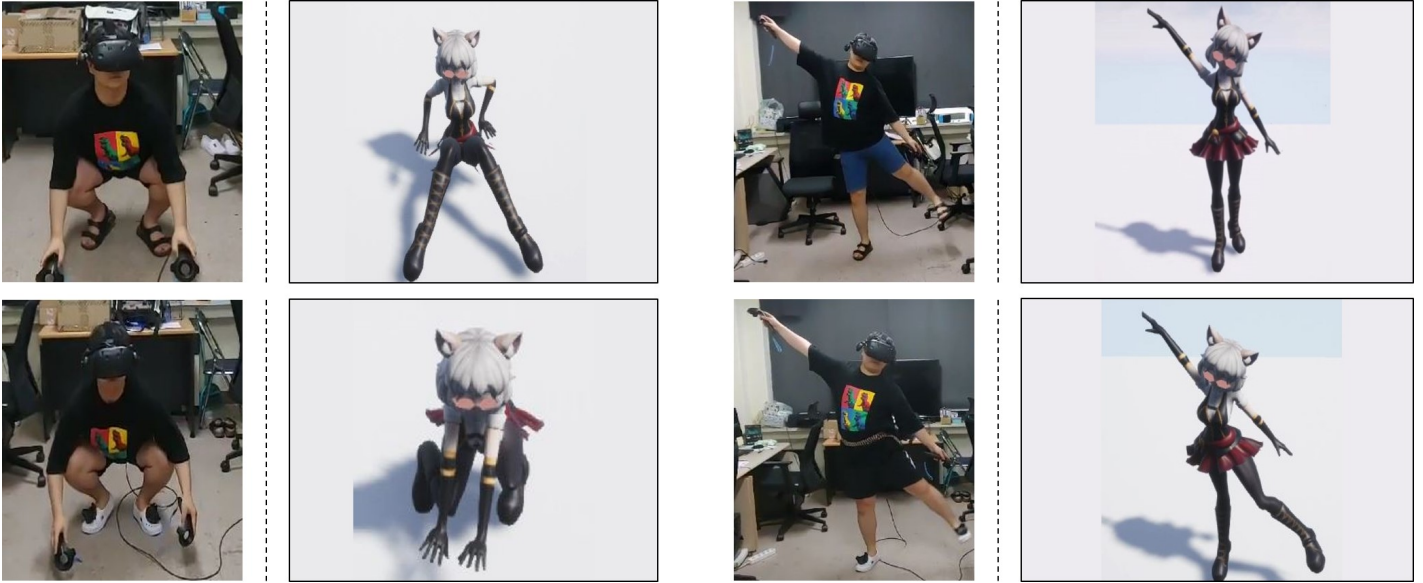


Figure 2. Comparison results with different numbers of motion sensors. (top) Using three motion sensors for head and both heads. (bottom) Using six motion sensors for pelvis, head, both hands, and both feet.

## 7. 결론 및 향후 연구

본 논문에서는 가상현실 게임을 위해 제작된 캐릭터 모델과 동작 데이터를 사용하여 일반적인 VR장비의 동작센서를 이용한 캐릭터 동작제어 방법을 제시하고 실험을 통해 유용성을 증명하였다. 일반적인 사람의 동작을 캡처한 모션캡처 데이터를 적용했을 때와 게임 제작을 위한 캐릭터의 애니메이션 데이터를 적용했을 때의 결과를 비교하여 게임 캐릭터 고유의 동작 스타일이 유지되었음을 증명하였다. 마지막으로 사용자의 허리와 양발에 동작 센서를 추가로 장착하는 실험을 통해 본 연구에서 제시하는 방법을 보다 정교하게 표현할 수 있다는 확장성도 증명하였다.

본 연구에서는 매 프레임마다 사용자가 착용한 입력 센서의 위치에 따라 캐릭터의 자세를 설정하므로 캐릭터와의 높은 인터랙션 반응 속도를 보장하기 위해 최소한의 캐릭터 보정만을 적용하였다. 따라서 사용자가 급격한 움직임을 보일 경우 캐릭터가 급격하게 변화하는 현상이 발생할 가능성이 있다. 이는 그래프 구조를 이용하는 등의 과거와 미래 캐릭터 동작 정보를 반영하는 방법으로 개선할 수 있을 것이다. 본 연구에서 사용한 방법의 단점인 급격한 움직임에 대한 부자연스러운 움직임 표현은 추후 연구를 통해 보완할 예정이다.

본 연구에서는 사용자 입력을 통해 상체 동작을 검색하여 그 동작과 유사한 동작을 캐릭터에 적용한다. 하지만, 게임의 현재 상황에 따라 각기 다른 동작을 수행해야 하는 경우가 있을 수 있다. 또한 상체 동작을 통하여 하체 동작을 유추할 수 있지만 동작 센서의 추가 없이 하체만을 직접 제어하는 것은 어려운 일이다. 그러나 추가적인 동작 센서의 장착은 게임을 플레이하는 사용자에게 불편함을 줄 수 있다. 따라서 이러한 단점을 보완하기 위하여 게임 상에서 게임의 현재 상황에 따라 다른 동작을 표현해야

하는 경우 혹은 달리기와 같은 하체의 움직임을 표현해야 하는 경우에 사용할 수 있는 상호작용 버튼을 추가하여 보다 원활한 게임이 진행되도록 할 수 있을 것이다.

## 감사의 글

본 연구는 문화체육관광부 및 한국콘텐츠진흥원의 2018년도 문화기술연구개발 지원사업으로 수행되었습니다.

## References

- [1] A. Vögele, B. Krüger, and R. Klein, "Efficient unsupervised temporal segmentation of human motion," in *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. Eurographics Association, 2014, pp. 167–176.
- [2] B. Krüger, A. Vögele, T. Willig, A. Yao, R. Klein, and A. Weber, "Efficient unsupervised temporal segmentation of motion data," *IEEE Transactions on Multimedia*, vol. 19, no. 4, pp. 797–812, 2016.
- [3] A. Aristidou, D. Cohen-Or, J. K. Hodgins, Y. Chrysanthou, and A. Shamir, "Deep motifs and motion signatures," in *SIGGRAPH Asia 2018 Technical Papers*. ACM, 2018, p. 187.
- [4] M. Müller and T. Röder, "Motion templates for automatic classification and retrieval of motion capture data," in *Proceedings of the 2006 ACM SIGGRAPH/Eurographics symposium*.



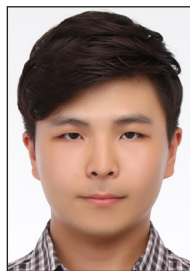
- sium on Computer animation. Eurographics Association, 2006, pp. 137–146.
- [5] Y. Wang and M. Neff, “Deep signatures for indexing and retrieval in large motion databases,” in *Proceedings of the 8th ACM SIGGRAPH Conference on Motion in Games*. ACM, 2015, pp. 37–45.
- [6] M. G. Choi and K. H. Lee, “Points-based user interface for character posing,” *Computer Animation and Virtual Worlds*, vol. 27, no. 3-4, pp. 213–220, 2016.
- [7] M. G. Choi and T. Kwon, “Motion rank: applying page rank to motion data search,” *The Visual Computer*, pp. 1–12, 2019.
- [8] J. Lee and S. Y. Shin, “A hierarchical approach to interactive motion editing for human-like figures,” in *Siggraph*, vol. 99, 1999, pp. 39–48.
- [9] J. Min, Y.-L. Chen, and J. Chai, “Interactive generation of human animation with deformable motion models,” *ACM Transactions on Graphics (TOG)*, vol. 29, no. 1, p. 9, 2009.
- [10] M. G. Choi, M. Kim, K. L. Hyun, and J. Lee, “Deformable motion: Squeezing into cluttered environments,” in *Computer Graphics Forum*, vol. 30, no. 2. Wiley Online Library, 2011, pp. 445–453.
- [11] R. Villegas, J. Yang, D. Ceylan, and H. Lee, “Neural kinematic networks for unsupervised motion retargeting,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8639–8648.
- [12] J. Lee, J. Chai, P. S. Reitsma, J. K. Hodgins, and N. S. Pollard, “Interactive control of avatars animated with human motion data,” in *ACM Transactions on Graphics (ToG)*, vol. 21, no. 3. ACM, 2002, pp. 491–500.
- [13] L. Kovar, M. Gleicher, and F. Pighin, “Motion graphs,” in *ACM SIGGRAPH 2008 classes*. ACM, 2008, p. 51.
- [14] J. Min and J. Chai, “Motion graphs++: a compact generative model for semantic motion analysis and synthesis,” *ACM Transactions on Graphics (TOG)*, vol. 31, no. 6, p. 153, 2012.
- [15] J. McCann and N. Pollard, “Responsive characters from motion fragments,” in *ACM Transactions on Graphics (TOG)*, vol. 26, no. 3. ACM, 2007, p. 6.
- [16] D. Holden, T. Komura, and J. Saito, “Phase-functioned neural networks for character control,” *ACM Transactions on Graphics (TOG)*, vol. 36, no. 4, p. 42, 2017.
- [17] M. G. Choi, K. Yang, T. Igarashi, J. Mitani, and J. Lee, “Retrieval and visualization of human motion data via stick figures,” in *Computer Graphics Forum*, vol. 31, no. 7. Wiley Online Library, 2012, pp. 2057–2065.
- [18] Y. Seol, C. O’Sullivan, and J. Lee, “Creature features: online motion puppetry for non-human characters,” in *Proceedings of the 12th ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. ACM, 2013, pp. 213–221.
- [19] Q. Ke, M. Bennamoun, S. An, F. Sohel, and F. Boussaid, “A new representation of skeleton sequences for 3d action recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 3288–3297.
- [20] W. Yoshizaki, Y. Sugiura, A. C. Chiou, S. Hashimoto, M. Inami, T. Igarashi, Y. Akazawa, K. Kawachi, S. Kagami, and M. Mochimaru, “An actuated physical puppet as an input device for controlling a digital manikin,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2011, pp. 637–646.
- [21] N. Numaguchi, A. Nakazawa, T. Shiratori, and J. K. Hodgins, “A puppet interface for retrieval of motion capture data,” in *Proceedings of the 2011 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. ACM, 2011, pp. 157–166.
- [22] J. Kim, Y. Seol, and J. Lee, “Realtime performance animation using sparse 3d motion sensors,” in *International Conference on Motion in Games*. Springer, 2012, pp. 31–42.
- [23] F. Wouda, M. Giuberti, G. Bellusci, and P. Veltink, “Estimation of full-body poses using only five inertial sensors: an eager or lazy learning approach?” *Sensors*, vol. 16, no. 12, p. 2138, 2016.
- [24] L. Guo and S. Xiong, “Accuracy of base of support using an inertial sensor based motion capture system,” *Sensors*, vol. 17, no. 9, p. 2091, 2017.
- [25] K. Amaya, A. Bruderlin, and T. Calvert, “Emotion from motion,” in *Graphics interface*, vol. 96. Toronto, Canada, 1996, pp. 222–229.
- [26] C. Rose, M. F. Cohen, and B. Bodenheimer, “Verbs and adverbs: Multidimensional motion interpolation,” *IEEE Computer Graphics and Applications*, vol. 18, no. 5, pp. 32–40, 1998.

- [27] A. Aristidou, Q. Zeng, E. Stavrakis, K. Yin, D. Cohen-Or, Y. Chrysanthou, and B. Chen, "Emotion control of unstructured dance movements," in *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. ACM, 2017, p. 9.
- [28] L. Hoyet, K. Ryall, K. Zibrek, H. Park, J. Lee, J. Hodgins, and C. O'sullivan, "Evaluating the distinctiveness and attractiveness of human motions on realistic virtual bodies," *ACM Transactions on Graphics (TOG)*, vol. 32, no. 6, p. 204, 2013.
- [29] M. Brand and A. Hertzmann, "Style machines," in *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*. ACM Press/Addison-Wesley Publishing Co., 2000, pp. 183–192.
- [30] E. Hsu, K. Pulli, and J. Popović, "Style translation for human motion," in *ACM Transactions on Graphics (TOG)*, vol. 24, no. 3. ACM, 2005, pp. 1082–1089.
- [31] I. Ismail, M. S. Sunar, H. W. Qian, and M. A. M. Arsad, "3d character motion deformation technique for motion style alteration," in *2015 4th International Conference on Interactive Digital Media (ICIDM)*. IEEE, 2015, pp. 1–4.
- [32] M. E. Yumer and N. J. Mitra, "Spectral style transfer for human motion between independent actions," *ACM Transactions on Graphics (TOG)*, vol. 35, no. 4, p. 137, 2016.
- [33] S. Xia, C. Wang, J. Chai, and J. Hodgins, "Realtime style transfer for unlabeled heterogeneous human motion," *ACM Transactions on Graphics (TOG)*, vol. 34, no. 4, p. 119, 2015.
- [34] D. Holden, I. Habibie, I. Kusajima, and T. Komura, "Fast neural style transfer for motion data," *IEEE computer graphics and applications*, vol. 37, no. 4, pp. 42–49, 2017.
- [35] I. Mason, S. Starke, H. Zhang, H. Bilen, and T. Komura, "Few-shot learning of homogeneous human locomotion styles," in *Computer Graphics Forum*, vol. 37, no. 7. Wiley Online Library, 2018, pp. 143–153.
- [36] M. G. Choi, S.-T. Noh, T. Komura, and T. Igarashi, "Dynamic comics for hierarchical abstraction of 3d animation data," in *Computer Graphics Forum*, vol. 32, no. 7. Wiley Online Library, 2013, pp. 1–9.

## 〈 저자 소개 〉

### 배태성

- 2014년~현재 가톨릭대학교 미디어기술콘텐츠학과 재학중
- 관심분야: 컴퓨터 그래픽스, 가상 현실, 게임 프로그래밍
- <https://orcid.org/0000-0002-1675-2464>



### 이은지

- 2016년~현재 가톨릭대학교 미디어기술콘텐츠학과 재학중
- 관심분야: 컴퓨터 그래픽스
- <https://orcid.org/0000-0003-3191-3892>



### 김하은

- 2017년~현재 가톨릭대학교 미디어기술콘텐츠학과 재학중
- 관심분야: 컴퓨터 그래픽스, 영상처리 등
- <https://orcid.org/0000-0002-5962-3160>



### 박민지

- 2008년 경북대학교 컴퓨터공학과 학사
- 2010년 서울대학교 컴퓨터공학부 석사
- 2013년~현재 티팟스튜디오(주) 대표이사
- <https://orcid.org/0000-0001-8814-5009>



### 최명걸

- 2004년 경북대학교 컴퓨터공학과 학사
- 2010년 서울대학교 컴퓨터공학부 박사
- 2011년 JST ERATO Design Interface Project 박사후 연구원
- 2012년~2013년 IPAB, Edinburgh University, 박사후 연구원
- 2014년~현재 가톨릭대학교 미디어기술콘텐츠학과 미디어공학전공 조교수
- 관심분야: 컴퓨터 그래픽스, 캐릭터 애니메이션, 가상현실
- <https://orcid.org/0000-0002-6089-9455>

