

IMU 센서기반 동작 인식을 위한 보정에 자유로운 데이터 매핑

김성훈^{*†} · 임양규^{*} · 김영호^{*} · 오나예^{*} · 채영호^{*}

^{*}중앙대학교 첨단영상대학원

Calibration free data mapping for IMU sensor based motion recognition

Seong Hun Kim^{*†} · Yang Kyu Lim^{*} · Young Ho Kim^{*} · Na Yea Oh^{*} · Young Ho Chai^{*}

^{*}Graduate School of Advanced Image Science, Multimedia and Film, Chung-Ang University

Key Words: Motion Capture(모션 캡처), Motion Simulation(동작 시뮬레이션), Inertial Sensor(관성 센서), Human Kinematics(인간 운동학), Algorithm(알고리즘)

Abstract

In order to obtain motion data of human movement in motion capture, a tool is needed to accurately measure the rotation value of each body part, such as the Initial Measurement Units (IMU). Simulation with this data requires the purchase of expensive software again. In addition to the cost burden, since the coordinate system of the 3D model of the 3D program and the character modeling are different from each other, it is difficult for the user who does not understand the 3D coordinate system to solve the problem. In this study, we have developed a method to simplify the steps up to the mapping of data and character modeling in case of motion capture using IMU sensor, and to develop an algorithm that allows the user to freely attach the sensor during the measurement process. At this stage, it is limited to a fixed position and it is normally applied to various modeling in the direction estimation and motion implementation, and it is confirmed that it is not affected by the sensor mounting direction.

1. 서 론

과거부터 게임이나 영화 산업 외에도 군사 훈련, 스포츠, 엔터테인먼트 등 여러 분야에서 실제 사람의 동작을 정확히 재현하기 위한 많은 시도들이 있었고, 현재에 이르러 모션 데이터를 3D 캐릭터 모델링에 적용하는 모션 캡처라는 기법으로 명명되었다. 모션 캡처의 방식에는 주로 Kinect, Leap Motion 등의 시각을 기반으로 하는 Visual tracking 방식, 몸에 센서를 부착하여 측정하는 Non-visual tracking 방식, 기계의 도움을 받아 측정하는 Robot-aided tracking 방식 등이 있다고 한다 [1]. 이 중 카메라 기반의 경우 빛과 주변 환경의 영향을 많이 받으며 마커 기반의 경우 높은 비용과 많은 계산량 등의 문제점이

있어 정확도와 효율성 측면에서 센서를 기반으로 하는 방식이 많이 진행되고 있다.

센서기반의 방식에서는 IMU 센서가 주로 사용되는데 센서와 소프트웨어를 함께 구매하는 경우 수백만원에서 수천만원의 비용이 요구되기에 소규모의 그룹이나 개인이 사용하기에는 많은 부담이 따랐고 이를 해결하기 위한 저가형 센서를 개발하는 등의 연구들이 진행되었다 [2]. 시뮬레이션 소프트웨어를 직접 구입하지 않는 경우라도 사용자가 올바른 동작 확인을 위해 직접 사용하는 센서의 좌표계, 시뮬레이션 프로그램의 좌표계, 모델링의 좌표계 및 3D 공간에서의 회전 계산까지 이해하고 있어야 하기 때문에 단순한 동작조차 직접 시뮬레이션하기 위해서는 매우 많은 시간과 노력이 필요로 하게 된다 [3-4]. 이런 단점과 더불어 기존 소프트웨어의 사용시 측정단계에서 요구되는 센서의 고정된 부착 방향이나 사람의 자세 등의 제한사항들이 사용자의 불편사항을 증가시키는 것으로 여겨진다.

†김성훈, 중앙대학교 첨단영상대학원

E-mail : tjdgns0258@cau.ac.kr

TEL : (02)812-7851

^{*} 중앙대학교 첨단영상대학원

본 연구에서는 모션 캡처에 대해 익숙하지 않은 사용자라도 기본 설정만으로 쉽게 시뮬레이션 해볼 수 있도록 복잡한 중간과정을 자동화하는 알고리즘을 개발하고 이를 통한 몇 가지 동작을 구현하여 올바르게 적용되는 것을 확인하였다.

2. 본론

2.1 Sensor 데이터

모션 캡처에 사용되는 IMU센서의 경우 3축 가속도계(Accelerometer), 각속도계(Gyroscope)로 구성되며 지자기 센서(Magnetometer)가 포함되기도 한다. 지구 중력과 자기장을 기준으로 3가지 센서에서 출력되는 데이터를 필터로 보정하여 물체의 회전 상태를 추정한다. 각 센서마다 기준으로 삼는 3축 좌표계 방향과 회전방식이 정의되어 있는데 실험에 사용된 ‘Xsens MTw Awinda’ 센서의 경우 전방을 X축, 위쪽을 Z축으로 하는 오른손 좌표계(Right-handed Coordinate System)를 사용한다. 또한 지자기 센서로부터 지구 자기장 방향을 추정할 수 있기 때문에 지구의 북쪽을 X축, 위쪽이 Z축, 서쪽이 Y축인 고정 좌표계를 정의하여 센서 좌표계와의 상대적인 회전 값 차이를 결과로 출력한다.

결과값으로 얻는 회전 데이터의 형식은 오일러 각(Euler Angles), 사원수(Quaternion), 회전 행렬(Rotation Matrix)을 주로 사용하는데 수학적 공식을 통해 서로 간의 변환이 가능하다. 오일러 각의 경우 Gimbal Lock이라는 널리 알려진 문제로 인해 정확한 회전 계산에 적합하지 않아 사원수 형식을 통한 회전 계산을 사용하였다. Sensor의 Global 좌표에 대한 상대적인 회전 값 데이터를 사원수 형식으로 얻으면 $q_{GS} = (q_0, q_1, q_2, q_3)$ 이며 이는 다시 $q_{GS} = (w, v)$ 로 표현이 가능하다.

2.2 회전 순서

3차원 공간에서 각도의 회전 방향이 시계 방향인지 반 시계 방향인지에 따라 왼손 좌표계 또는 오른손 좌표계로 나뉜다. 또한 회전하는 축의 순서에 따라 여러 조합으로 나뉘는데 오일러 각에서는 X, Y, Z 회전 순서 12가지를 ‘Proper Euler angles’ 과 ‘Tait-Bryan angles’ 2가지 부류로 분류한다. Tait-Bryan 방식에서의 X, Y, Z축을 기준으로 한 각 회전은 Roll, Pitch, Yaw 라고도 표현하며 오일러 각의 회전 순서를 기준으로

사원수와 회전 행렬로의 변환 계산식 또한 다르기 때문에 각 순서에 맞는 정확한 공식을 사용해야 한다. 연구에 사용된 센서는 반 시계 방향으로 XYZ 순서에 따라 X축이 먼저 회전하며 사원수 형식으로 얻은 회전 값을 다시 오일러 각으로 변환하기 위한 공식은 다음과 같다.

$$\begin{aligned} \text{Roll} &= \tan^{-1} \left(\frac{2q_2q_3 + 2q_0q_1}{2q_0^2 + q_3^2 - 1} \right) \\ \text{Pitch} &= -\sin^{-1}(2q_1q_3 - 2q_0q_2) \\ \text{Yaw} &= \tan^{-1} \left(\frac{2q_1q_2 + 2q_0q_3}{2q_0^2 + q_1^2 - 1} \right) \end{aligned} \quad (1)$$

2.3 시뮬레이션 프로그램

센서로부터 얻은 데이터를 사용하여 올바른 동작이 구현되는지 확인하기 위해서는 3D 모델링에 데이터를 매핑하여 확인하여야 하며 Unreal, Unity, Blender, OpenGL 등의 소프트웨어 또는 자체적으로 개발하는 프로그램이 사용된다. 이때 각 소프트웨어가 모두 다른 좌표계 시스템을 갖고 있으며 사용되는 모델링 또한 각 부위마다 연결된 다른 부위로부터 영향을 받아 좌표계가 다를 수 있기 때문에 최종적으로 센서와 소프트웨어, 모델링 3종류의 좌표계를 고려하여 정확히 계산하여야 데이터의 올바른 동작을 확인할 수 있다. 실험에 사용한 Unity의 경우 Y축이 위쪽, Z축이 정면, X축이 오른쪽을 향하는 왼손 좌표계 방식을 사용하며 회전 방향 또한 센서와 반대인 시계방향으로 회전한다. 회전 순서 또한 ZXY 순서로 XYZ순서로 회전하는 센서와 다르게 적용된다.

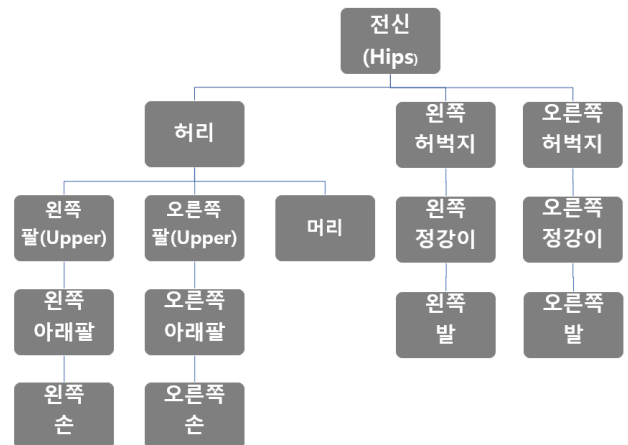


Fig. 1 Human body part hierarchy tree

2.4 캐릭터 모델링

데이터를 적용할 모델링을 불러오면 여러 신체부위가 계층 구조에 따라 연결되어 있으며 각각 다른 좌표계 상태를 가지고 있음을 확인 할 수 있다. 이는 사용하는 모델링 마다 중간에 다른 부위가 연결되어 누적된 회전 값이 다르기 때문이며 Fig. 2처럼 같은 자세를 취하고 있더라도 각 부위의 좌표 상태는 모델링 마다 다를 수 있다. 예를 들어 다리의 움직임의 경우 허벅지를 움직이면 하위 구조에 연결되어 있는 종아리, 발 부위 또한 회전의 영향을 받게 되는데 이때 각 부위의 지역 좌표계 상태는 변하지 않았더라도 전역 좌표계를 기준으로 본 회전 값이 달라지기 때문에 각 부위에 연결되어 있는 상위 부위의 회전 값들을 고려해야 하는 것이다. 따라서 상위의 연결된 부위들이 많을수록 계산이 복잡해지다 보니 모든 뼈 부위를 추적하는 방법 대신 주요 움직임에 관여하는 일부 신체 부위들을 인체 구조에 따라 단순화하려는 노력들이 진행되어왔다 [5]. 본 연구에서는 사전에 Fig.1의 형태로 모델링 및 애니메이션에 자주 쓰이는 신체 부위를 단계별로 나누었다. 예를 들어 종아리 부위는 전신, 허벅지의 상위 2단계로부터 영향을 받고 발 부위는 전신, 허벅지, 종아리 상위 3단계의 영향을 받는다.

센서로부터 측정한 부위의 데이터를 얻은 후에 모델링의 각 부위에 연결하여 원하는 동작을 구현하기 위해서는 해당 자세에서 다음 동작을 위해 회전해야 하는 올바른 방향을 계산해야 하며 이를 다시 현재 모델링의 좌표계 상태에 맞게 변환하여 대입시키는 과정이 필요하다.

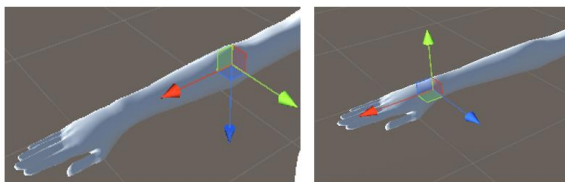


Fig. 2 axis difference between right lower arm and hand.

2.5 좌표계 변환 과정

센서로부터 얻어지는 q_{GS} 를 통해 우리는 각 상태에서의 회전 값을 알 수 있다. 따라서 A 상태의 센서를 C 상태로 회전시키는 단위 사원수 값 q_B 가 있다고 할 때 다음 수식이 성립하며

$$q_C = q_B * q_A \quad (2)$$

$$q_B = q_C * q_A^* \quad (3)$$

결레 사원수인 q_A^* 는 q_A 의 역과 같기 때문에 q_B 를 계산할 수 있다.

q_B 로부터 다음 상태로 변환시키는 회전 방향을 얻었으나 이는 센서로부터 지구를 기준으로 하는 q_{GS} 의 값이기 때문에 Unity에 적용하기 위해서는 Unity 전역 좌표계를 기준으로 변환해야 한다. 따라서 센서의 전역 좌표계와 Unity 전역 좌표계를 일치시키는 계산이 필요하고 두 전역 좌표계의 전방과 위쪽 좌표계를 동일하게 놓는 것으로 간주하여 계산한다. 이때 센서의 전방인 X축은 Unity에서 Z축, 위쪽 방향인 Z축은 Unity Y축과 같으며 나머지 Y축이 Unity의 X축과 반대인 것에 주의한다. 이를 수식으로 표현하면 Global 좌표를 Unity 좌표로 변환시키는 Matrix M_{UG} 가 다음과 같다.

$$M_{UG} = \begin{pmatrix} 0 & -1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}^{\text{Global}} = \begin{pmatrix} -Y & Z & X \end{pmatrix}^{\text{Unity}} \quad (4)$$

식 (3)으로부터 얻은 회전 방향을 식 (4)의 M_{UG} 를 사용하여 올바른 회전방향을 구할 수 있는데

$$q_{US} \cdot v = M_{UG} * q_{GS} \cdot v, \quad q_{US} \cdot w = -q_{GS} \cdot w \quad (5)$$

q_{GS} 의 벡터부분을 M_{UG} 와 곱하여 좌표계를 변환하고 회전방향이 기존 반 시계 방향에서 방향으로 변하였기 때문에 반대 방향으로 바꾸어 준다.



Fig. 3 Experiment with random attachment of IMU sensor. Up: case 1, Down: case 2

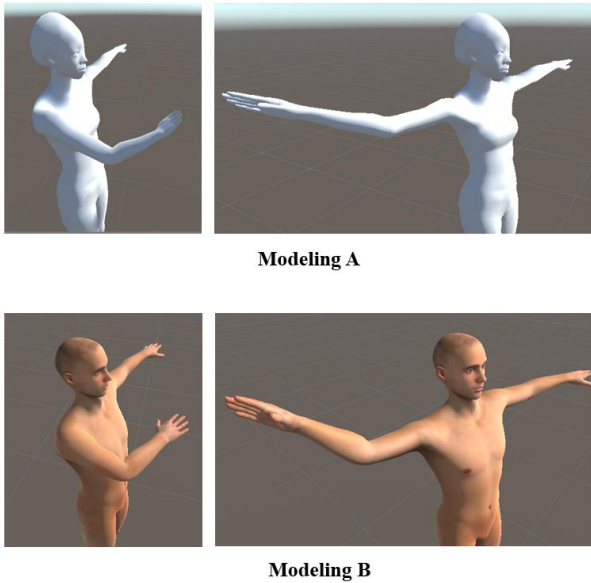


Fig. 4 Example of two models using case 1&2 data

마지막으로 모델링의 각 부위별 회전 값 차이를 고려해야 하며 이 값을 q_{PU} 라고 할 때 전역 좌표계 기준으로 상대적인 회전 값 차이를 계산한다.

$$q_{PU} = (\text{현재 부위의 회전 값 } q_P) * (\text{상위 부위의 회전 값 } q_P^{-1}) \quad (6)$$

식 (6)으로부터 얻은 각 부위의 q_{PU} 값들을 계층구조에 따라 곱해주어야 연속적으로 변화하는 동작을 제대로 구현할 수 있다. 예를 들어 종아리 부위는 $q_{PU}(\text{종아리}) * q_{PU}(\text{허벅지}) * q_{PU}(\text{전신})$ 순서의 3번의 사원수 곱셈이 필요하고, 하위 부위의 발을 추적하는 경우 $q_{PU}(\text{발}) * q_{PU}(\text{종아리}) * q_{PU}(\text{허벅지}) * q_{PU}(\text{전신})$ 순서로 4번의 곱셈 과정이 필요하다. 위 과정을 거쳐서 얻은 q_{PU} 를 식 (5)의 q_{US} 와 곱하면 센서로부터 모델링 부위의 좌표계에 대응하는 q_{PS} 를 얻을 수 있다.

2.6 알고리즘 적용 실험

센서의 데이터 측정에 앞서 사람은 항상 T 자세에서 시작하여 동작을 수행하고 캐릭터 모델링의 초기 자세 또한 T 자세로 동일 하게 설정하였다. 사용되는 센서의 개수는 허리 부위에 고정적으로 1개가 필요하며 이후 원하는 부위 마다 한 개씩 추가되었으며 이번 실험에서는 허리, 위 팔, 아래 팔, 손 부위에 총 4개가

사용되었다. 센서 부착 시 사람의 정면 방향을 찾기 위해 허리에는 가로 방향으로 가정하여 부착하였으며 나머지 3부위는 Fig. 3처럼 방향에 관계없이 무작위로 부착하였다. 특정 방향으로 오류가 있는지 확인하기 위해 360도 사방으로 팔을 5번씩 돌리며 데이터를 측정하였다. 이후 실험을 진행하여 얻은 두가지 데이터를 다시 다른 좌표계 상태를 가지고 있는 2개의 모델링에 각기 적용하여 총 4가지의 결과를 비교하였으며 Fig. 4의 그림에서 보여지는 것처럼 정상적으로 실제 동작과 유사하게 나타나는 것을 확인할 수 있었다.

4. 결 론

대부분의 3D 모델링들은 비슷한 계층구조를 가지며 각 부위들이 연결되어 있으나 개수나 구조 자체가 모두 동일하지는 않기 때문에 부위마다 다른 좌표계 상태를 가지고 있다. 여기에 다시 각 센서와 소프트웨어마다 달리 적용되는 좌표계 시스템을 고려하여 계산하여야 하니 별도의 소프트웨어를 통해 시뮬레이션을 하기 위해서는 많은 작업이 필요하다.

본 연구는 IMU센서를 사용하여 모션 캡처를 진행하는 경우 시뮬레이션 단계에서 필요한 데이터와 모델링의 좌표계 변환 문제를 해결하고자 진행되었다. 위치가 변하지 않는 제한적인 조건에서의 실험을 통해 센서의 부착방향에 상관없이 측정한 데이터를 임의의 모델링에 연결하여 올바르게 동작하는 것을 확인할 수 있었다. 향후 연구를 통해 동적인 상황에서의 위치 추적을 비롯한 더욱 정밀한 동작을 구현할 수 있는 알고리즘으로 발전시킬 예정이며, 이러한 알고리즘 제공을 통해 사용자가 센서와 소프트웨어에 종속적이지 않게 손쉽게 모션 캡처를 수행할 수 있을 것으로 예상된다.

후 기

이 논문은 2018년도 과학기술정보통신부의 재원으로 정보통신기술진흥센터의 SW스타랩 지원(2018-0-00599)과 국토교통부의 재원으로 국토교통과학기술진흥원의 지원(18CTAP-C132982-02)을 받아 수행된 연구임.

참고문헌

- [1] Zhou, Huiyu & Hu, Huosheng, 2008, “Human motion tracking for rehabilitation—A survey”, *Biomedical Signal Processing and Control*, Vol. 3, No. 1, pp. 1-18.
- [2] McCarron, Brandon., 2013, "Low-cost imu implementation via sensor fusion algorithms in the arduino environment."
- [3] Roetenberg, Daniel, Henk Luinge, and Per Slycke., 2009, "Xsens MVN: full 6DOF human motion tracking using miniature inertial sensors.", *Xsens Motion Technologies BV, Tech. Rep.*
- [4] Arsenault, Dennis., 2014, “A quaternion-based motion tracking and gesture recognition system using wireless inertial sensors.”, Diss. Carleton University.
- [5] Saini, Sanjay, et al., 2015, "Markerless human motion tracking using hierarchical multi-swarm cooperative particle swarm optimization.", *PLoS One* 10.5, e0127833.