

## 키넥트 카메라를 이용한 관절 가동 범위 측정과 보정

\*정주현 \*\*윤명석 \*\*\*김상준 \*박구만

서울과학기술대학교 \*전자IT미디어공학과, \*\*일반대학원 미디어IT공학과,  
\*\*\*나노IT디자인융합대학원 정보통신미디어공학전공

\*churyel10@naver.com

## Joint Range Measurement and Correction using Kinect Camera

\*Jeong, Juheon \*\*Yoon, Myeongsuk \*\*\*Kim, Sangjoon \*Park, Gooman

\*Dept. of Electronics and IT Media Engineering,

\*\*Dept. of Media IT Engineering,

\*\*\*Dept. of Information Technology and Media Engineering,

Seoul National University of Science and Technology

### 요약

가상현실과 증강현실의 대중화로 사람의 동작을 실시간 3D 애니메이션으로 구현하는 연구가 활발히 진행 중이다. 특히 Microsoft에서 키넥트(Kinect)를 개발함에 따라 저렴한 가격에 부가적인 장치 필요 없이 간단한 조작만으로도 3D 모션 정보 취득이 가능해졌다. 하지만 키넥트 카메라는 마커 기반 모션 캡처 시스템에 비해 관절 정보의 추정 성능이 뒤떨어져 낮은 정확도를 보이는 단점을 지니고 있다. 이에 본 논문에서는 키넥트 카메라를 이용해 사람의 관절 정보를 취득하고 이것에 관절 가동 범위(Range of Motion, ROM)를 적용하여 비정상적인 동작을 보정하는 시스템을 제안한다. ROM을 구하는 방법으로는 수행자가 모든 관절에 대해 회전 운동을 수행한 뒤 관절들의 회전 운동 정보를 취득, 분석하여 정상적인 ROM을 설정하고 실험으로부터 사람의 동작이 개선되는 것을 확인하였다.

### 1. 서론

오늘날 디지털 영상 장비와 영상 처리 기술의 발전으로 확장현실(Extended Reality, XR)과 같은 3D 콘텐츠 보급이 증가함과 동시에 현실 세계의 사람의 동작을 가상 세계의 애니메이션으로 정확하게 구현시키는 기술에 대한 연구가 활발히 진행 중이다. 모션 캡처 시스템의 대표적인 예로는 크게 마커 기반의 기계식과 자기식, 광학식 시스템이 있으며 취득된 모션 정보를 보정하기 위한 연구로 관절 좌표에 칼만(Kalman) 필터, RANSAC(Random Sample Consensus) 알고리즘을 적용하거나 회전 값을 쿼터니언 좌표계로 변환, 또는 신체 길이의 제약이나 데이터베이스의 사전 정보를 활용하여 관절 좌표를 맵핑하는 것과 같이 다양한 방법이 제안되었다[1,2,3]. 또한 최근 Microsoft에서 개발한 키넥트(Kinect) 카메라는 Fig. 1과 같이 RGB 카메라와 적외선 센서로부터 신체의 관절 정보를 제공함에 따라 비콘(Vicon)과 같은 다른 마커 기반 모션 캡처 시스템들과 달리 설비의 부담과 전문적인 프로그램 활용 수고 없이도 실시간 3D 애니메이션 구현이 가능해졌다[4,5,6]. 하지만 키넥트는 다른 시스템에 비해 관절의 위치 추정 성능이 현저히 낮아 신체의 움직임이 정상적인 관절 가동 범위(Range of Motion, ROM)를 넘어서는 추정 오류를 유발할 확률이 매우 높고 기존 연구의 보정 방법으로는 이를 해결하는데 큰 어려움이 있다. 여기서

ROM이란 Fig. 2와 같이 각 관절이 자신의 모양새와 근육, 건, 인대의 해부학적 구조에 의해 갖게 되는 고유한 정상 동작 범위를 뜻하는 것으로 이에 기반한 관절의 회전 운동이 신체의 움직임을 발생시킨다[7].



Fig. 1 Color and depth image of Kinect.

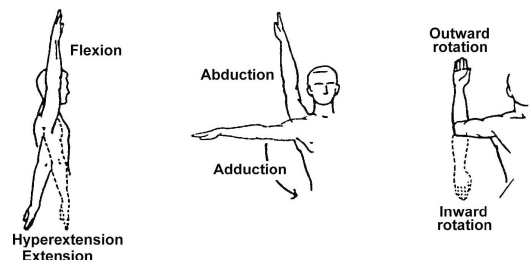


Fig. 2 Range of arm joint motion[8].

따라서 본 논문에서는 키넥트를 이용한 모션 캡처 시스템에서의 관절 추정 오류를 보정하고자 15개의 관절에 대해 회전 운동을 시행한 뒤 키넥트 카메라로부터 관절 회전 정보를 취득 및 분석해 직접 ROM을 설정하고 실제 데이터에 이를 적용해 관절의 비정상적인 회전 운동을 제거하여 신체의 자연스러운 움직임을 구현하는 알고리즘을 제안한다.

## 2. 본론

ROM을 구하기 위해 우선 수행자는 키넥트 카메라 앞에서 Fig. 3과 같이 15개의 관절마다 xy평면, xz평면, yz평면에 대해 최대한으로 가능한 회전 운동을 시행한다. 이때 키넥트 카메라에서는 센서를 통해 수행자의 동작을 인식하고 매 프레임  $t$ 마다 해당 관절들  $j$ 을 추적하여 각각의 3차원 상 이동 및 회전 정보  $T(x_t^j, y_t^j, z_t^j)$ ,  $R(w_t^j, x_t^j, y_t^j, z_t^j)$ 을 제공해준다.

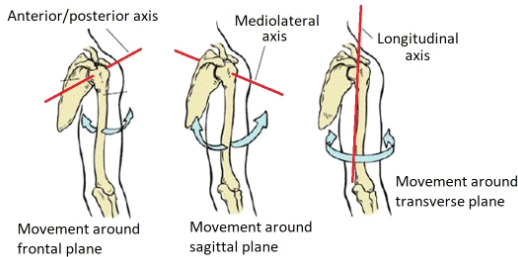


Fig. 3 Arm joint movements on the 3D plane[9].

다음으로 저장된 동작 정보는 FBX로 넘겨진다. 여기서 FBX란 3D 데이터 형식으로 이를 이용하여 관측된 골격의 형태를 유추하고 레이어들과 관절들의 상속 관계, 각 관절의 위치 및 회전 정보를 지정하여 형태의 움직임을 구현할 수 있다[10]. 이때 회전 값  $R(w_t^j, x_t^j, y_t^j, z_t^j)$ 은 관절 간 부모-자식 상속 관계를 통해 Fig. 4와 같이 전역 값(global value)에서 지역 값(local value)으로 변환되는데 이는 관절의 회전 운동 구현에 최종적으로 이용되는 값이다. 회전 값  $R(w_t^j, x_t^j, y_t^j, z_t^j)$ 은 Fig. 5와 같이 4차원의 쿼터니언(Quaternion) 좌표계  $Q(w, x, y, z)$ 에서 3차원 상 xyz 세 개의 축으로 이루어진 오일러(Euler) 좌표계  $E(x, y, z)$ 로 변환되며 XYZ 순서로 회전 운동을 구현한다.

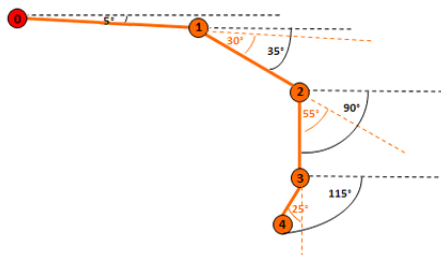


Fig. 4 Global and local rotations of joint hierarchy[11].

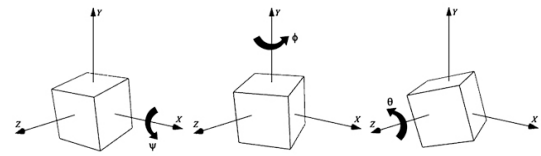


Fig. 5 Euler rotation in the 3D axis[12].

관절별 시간축 상 오일러 회전각  $R(x_t^j, y_t^j, z_t^j)$ 의 최대, 최솟값을 분석하여 ROM을 구한다. 관절 회전 값  $R(x_t^j, y_t^j, z_t^j)$ 의 정상 범위는  $E(x, y, z) \in (-\pi, \pi)$  성질상 값의 부호가 반전되는 것을 고려하여 최대 3개의 복수구간이 되도록 설정해 Table. 1에 나타내었다.

|            |   | Range 1 |      | Range 2 |     | Range 3 |     |
|------------|---|---------|------|---------|-----|---------|-----|
|            |   | Min     | Max  | Min     | Max | Min     | Max |
| Spine base | X | -180    | -100 | 138     | 180 |         |     |
|            | Y | -35     | 44   |         |     |         |     |
|            | Z | -180    | -150 | 142     | 180 |         |     |
| Spine mid  | X | 0       |      |         |     |         |     |
|            | Y | -10     | 6    |         |     |         |     |
|            | Z | 0       |      |         |     |         |     |
| Neck       | X | -180    | -140 | 150     | 180 |         |     |
|            | Y | -15     | 5    |         |     |         |     |
|            | Z | -180    | -172 | 165     | 180 |         |     |
| Shoulder   | X | -5      | 18   |         |     |         |     |
|            | Y | -3      | 3    |         |     |         |     |
|            | Z | -136    | -92  |         |     |         |     |
| Arm        | X | -180    | -124 | -64     | 180 |         |     |
|            | Y | -88     | 28   |         |     |         |     |
|            | Z | -180    | 18   | 74      | 180 |         |     |
| Elbow      | X | -180    |      | 0       |     | 180     |     |
|            | Y | -90     | -10  | 34      | 90  |         |     |
|            | Z | -158    | -25  | 44      | 165 |         |     |
| Hip        | X | -92     | -68  | 8       | 175 |         |     |
|            | Y | -88     | 8    |         |     |         |     |
|            | Z | -90     |      | 48      | 162 |         |     |
| Knee       | X | -180    |      | 0       |     | 180     |     |
|            | Y | -15     | 48   |         |     |         |     |
|            | Z | 2       | 128  |         |     |         |     |
| Ankle      | X | -70     | 34   |         |     |         |     |
|            | Y | 65      | 82   |         |     |         |     |
|            | Z | -59     | -18  |         |     |         |     |

Table. 1 Maximum and minimum values of Euler angles (degree) per joint in multiple sections.

최종적으로 키넥트 카메라로 촬영한 실험 데이터의 관절 회전 값  $R(x_t^j, y_t^j, z_t^j)$  중 하나라도 해당 구간을 벗어날 경우 이전 프레임의 값  $R(x_{t-1}^j, y_{t-1}^j, z_{t-1}^j)$ 으로 대체함으로써 ROM을 구현한다. Fig. 6 (Up)은 좌표 평면에 표시된 임의의 관절에서의 시간축에 대한 회전 값이며 (Down)는 이것에 ROM이 적용된 결과를 표시한 것이다. 특정 시간 동안 회전 값이 비정상적으로 작아지자 이전 시간의 회전 값으로 대체하여 값이 정상 범위 내에 유지되도록 하였다.

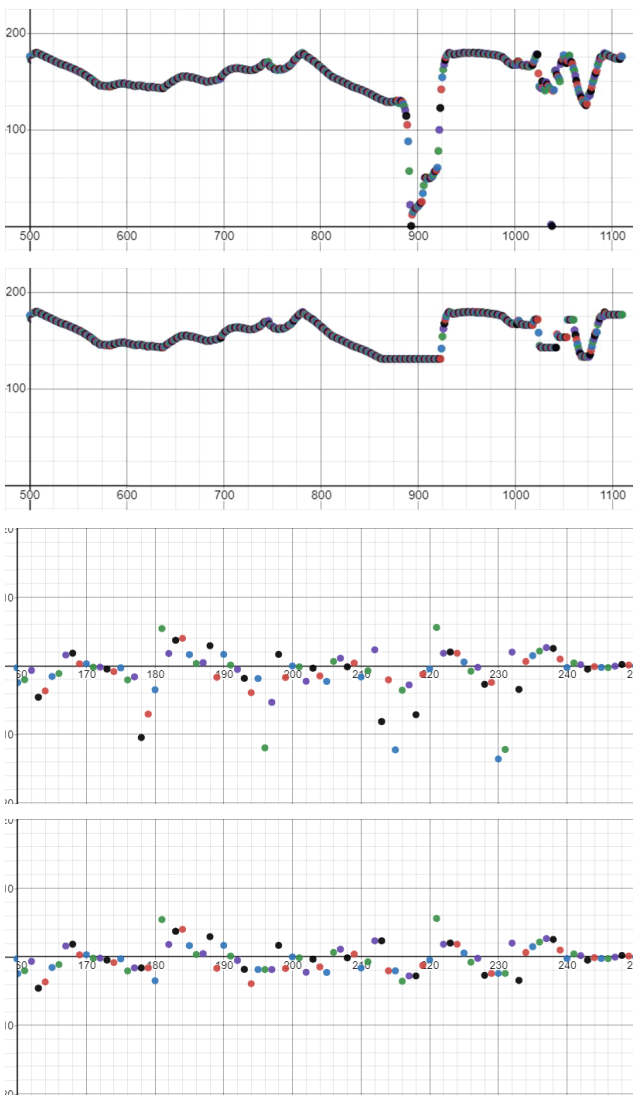


Fig. 6 (Up) Graph of source 1D coordinate and (Down) graph of result 1D coordinate applied ROM.

### 3. 실험 결과

본 연구에서 관절 회전 좌표 추출은 Kinect SDK 2.0를 이용했으며 ROM 알고리즘 개발은 C/C++를, 그리고 실시간 3D 애니메이션 형성 및 표출은 FBX SDK 2019, Open Frame Work 0.9.8로 자체 개발된 프로그램을 이용하였다.

본 실험은 키넥트 카메라로 촬영한 동작에 ROM을 적용하여 자세 및 동작이 개선되는지를 확인하기 위한 것으로 5명이 실험에 참가하여 시스템을 실행시키고 키넥트 카메라 앞에서 임의의 동작을 수행하였다. 실험 결과 Fig. 7 (Left)는 GUI에 표출된 원본 형태이며 (Right)는 이것에 ROM이 적용된 결과를 표출한 것이다. ROM을 적용하여 비정상적인 회전 운동을 보이는 목, 어깨 그리고 팔꿈치 관절을 보정해 피실험자의 동작이 더욱 자연스럽게 실제와 유사한 형태가 되었다.

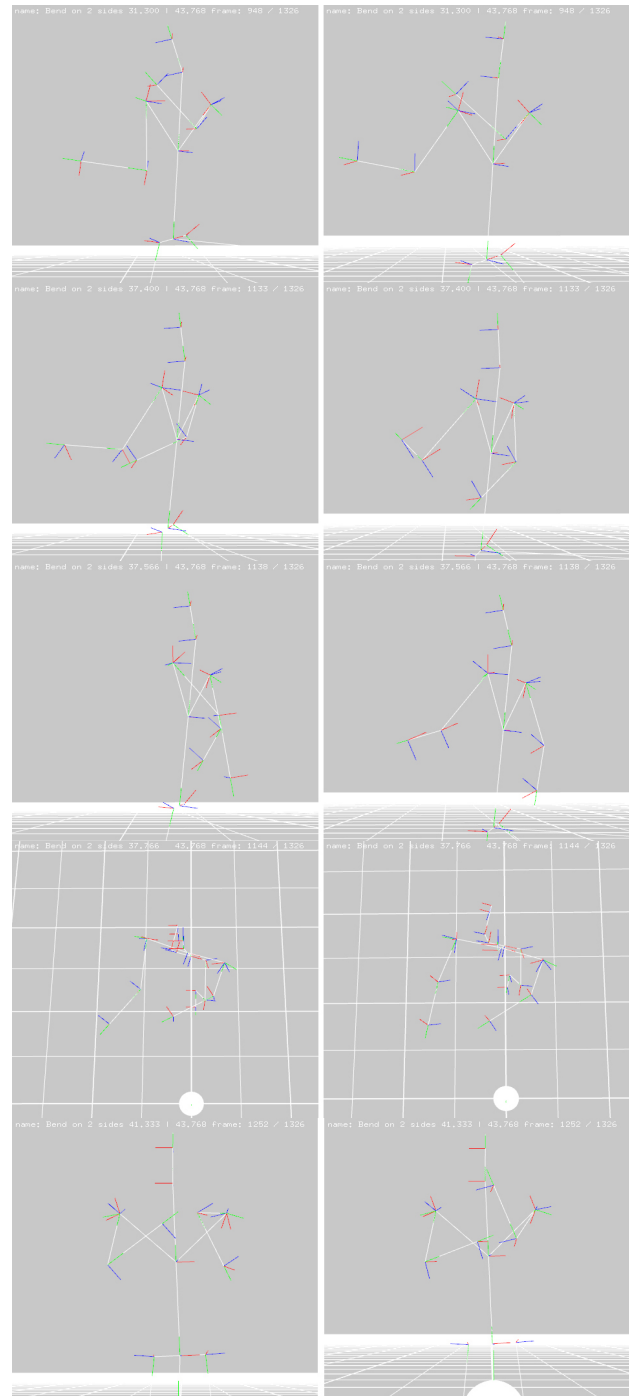


Fig. 7 (Left) Source 3D motion on GUI and (Right) result 3D motion applied ROM on GUI.

### 4. 결론 및 향후 연구 방향

키넥트로 실제 사람의 동작을 가상의 애니메이션으로 구현하기 위한 연구가 활발히 진행 중이다. 그러나 키넥트 카메라는 관절 회전 정보 추정 성능에 문제가 많아 추가적인 보정 알고리즘을 필요로 한다. 이를 해결하기 위하여 본 논문에서는 ROM을 구한 뒤 키넥트 카메라에서 전달되는 신체 골격의 관절 회전 정보에 이를 제약으로 설정하여 부정확한 자세 및 동작을 개선하는 알고리즘을 제안하였다. 실험결과

순간적으로 ROM을 벗어난 관절을 감지하고 관절의 회전 값을 설정된 ROM 이내가 되도록 보정하여 동작이 더욱 자연스러워지는 것을 확인하였다.

차기 연구로는, 본 연구의 통계에 근거한 알고리즘의 단점을 보완하고자, 키넥트 센서의 깊이 및 RGB 정보를 활용하여 신체 부위의 길이와 방향에 제약을 설정하는 방식으로 관절을 보정하는 알고리즘을 구현하는 것을 목표로 한다.

## 감사의 글

본 연구는 2019년도 학연연계 사업화 선도모델 지원사업으로 수행되었음. (홍릉 Tech-Biz OPERA 센터구축을 통한 학연연계 기반의 산학연 협력체계 구현과제)

## 참고 문헌

- [1] Hao Liy, "Mobility Fitting using 4D RANSAC", Eurographics Symposium on Geometry Processing, 2016.
- [2] Li Ji, "Motion Trajectory of Human Arms based on the Dual Quaternion with Motion Tracker", Springer Science Business Media, New York, 2015.
- [3] Hyunjoon Shin, "Computer Puppetry: An Importance-Based Approach", ACM Transactions on Graphics, Vol. 20, No. 2, pp. 67 - 94, 2016.
- [4] Sungho Kim, "Realtime 3D Human Full-Body Convergence Motion Capture Using a Kinect Sensor", Journal of Digital Convergence, Vol. 14, No. 1, pp. 189-194, 2016.
- [5] Iksu Kim, "Development of a Human Animation Viewer Using a Kinect Sensor", Korea Computer Congress, pp. 1093-1095, Korea, 2013.
- [6] "Comparison of Kinect v1 and Kinect v2", last modified Jun 7, 2018, accessed Oct 17, <https://design486.tistory.com/460>.
- [7] "Range of motion", last modified Jun 1, 2017, accessed Oct 17, <https://m.blog.naver.com/kimpulse/221019431886>.
- [8] "Range of motion", last modified 2015, accessed Oct 17, [https://brooksidepress.org/basic\\_patient\\_care](https://brooksidepress.org/basic_patient_care).
- [9] "Axis of the human body", last modified Jan 12, 2019, accessed Oct 17, <https://nusantarafood.me>.
- [10] "Loading mesh of FBX file", last modified Apr 10, 2010, accessed Oct 17, <https://blog.naver.com/lifeisforu/80105357896>.
- [11] "Local and global joint rotation", last modified 2016, accessed Oct 17, <https://software.intel.com/sites/landingpage/realsense/camera-sdk/v1.1/documentation/htm>.
- [12] "Quaternions", accessed Oct 17, <http://what-when-how.com/advanced-methods-in-computer-graphics>.