

증강현실을 이용한 모션 인식 시스템

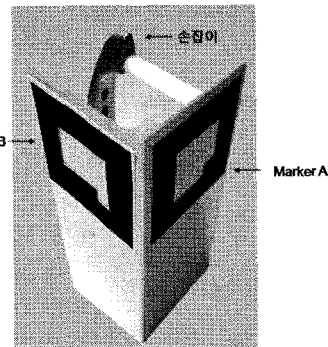
*남일호, 선문대학교 컴퓨터공학부, aseitasi@sunmoon.ac.kr
 **유영기, 선문대학교 정보통신공학부, ryu@sunmoon.ac.kr

The System of Motion Recognition using Augmented Reality

*Il-ho Nam, Department of Computer Engineering, Sunmoon University
 **Young-kee Ryu, Department of Information and Communication Engineering, Sunmoon University

Abstract - Motion capture system that is used in movies and games today requires large equipments and expensive that it is still not common as a reasonable family game. Therefore, this research project focused on solving these problems mentioned above and move on to develop a new type of approach and by using reasonable family use USB camera and augmented reality to develop a bodily sensation motion capture system which can be used in making a video game. In order to develop this, I used a pair of Marker which can be placed on both hands and one USB camera and the image that was extracted in real time goes through the AR toolkit and abstracted the 3D coordinates of data and developed a motion capture system.

사용자가 주먹을 쥐고 자신의 가슴 앞쪽에 위치하도록 하는 것을 기본 상태로 하였다. 본 시스템에서 4가지 모션을 모두 인식하기 위해서는 마커 하나를 이용하는 것으로서는 불가능하여, 다른 패턴의 마커를 90°시계방향에 위치하도록 하였다.



<그림 2> Marker 두 개를 한 쌍으로 포함하는 손잡이

1. 서 론

현재 비디오 게임분야에서는 최근 체감형, 몰입형 게임들이 두각을 나타내고 있는데, 체감형 게임이란 모션플랫폼을 이용한 시뮬레이터형 게임기로서, 1인용부터 다인용에 이르기까지 다양한 시스템을 구축할 수 있는 것을 말한다. 게임 상에서 현실감을 느끼게 하는 것이 가장 중요한 요소로 작용하기 때문에 가상현실 기술들이 주목되고 있어 게임기를 조작하는 사용자들은 더욱 생생한 느낌을 전달받을 수 있게 되었다. 최근 주목 받고 있는 것이 Nintendo Wii[1]이다. 하지만, 현재 영화나 게임에 이용되는 모션캡처 시스템은 장비가 크고 상당히 고가이기 때문에 간단한 가정용 게임에는 적용하기 어렵다.

따라서 본 논문에서는 이러한 문제를 해결한 새로운 접근방법을 개발하고자 가정용 USB camera와 증강현실을 이용하여 Video 게임이 가능한 체감형 모션 인식 시스템을 개발하기 위하여 양손에 부착이 가능한 Marker 한 쌍과 USB camera 1대를 이용하여, 추출된 실시간 이미지에서 Marker의 정보를 ARToolkit[2] 통해 3차원 좌표 데이터를 추출하여 모션 인식이 가능한 System을 개발하고자 한다.

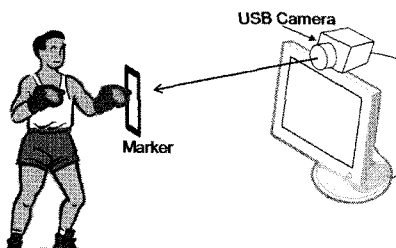
2. 본 론

2.1 증강현실 기술

현재 증강현실을 구현하기 위하여 사용되는 라이브러리는 대표적으로 ARToolkit과 Open Scene Graph[3]이 있다. 이것의 기본적인 역할은 지정된 marker를 찾기 위해 USB camera로부터 얻어 온 live video image를 binary image로 변환 후 모든 square region을 탐색한다. 탐색 되어진 square내의 pattern을 capture하여 지정된 marker와 match를 시도하여 실제 marker인지를 결정하고 marker의 square size와 pattern orientation을 사용하여 real video camera의 position을 계산한다. 실제 camera의 계산 후, virtual object들을 image에 그릴 수 있다.

2.2 시스템의 구성

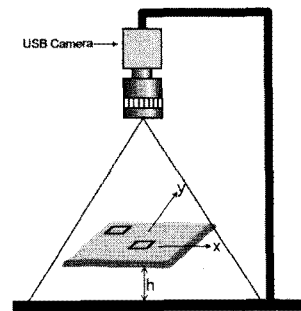
본 시스템에서는 USB 카메라 앞에서 마커를 손 앞에 위치하도록 한다. 손의 움직임에 따라 같이 움직이는 마커의 움직임을 USB 카메라가 인식한 후 마커에 대한 정보를 화면에 다시 보여주고 있다. 이 정보로부터 상대위치, 카메라로부터의 거리, x·y·z 축에 대한 회전을 구할 수 있었다. 모션을 판단하는 종류를 총 4가지로 하였다. 이 모션은 또한 복성의 편차 종류로 하여 각각의 모션에 대한 뚜렷한 다른 점을 바탕으로 하였다.



<그림 1> USB CAMERA와 Marker로 구성된 모션캡처 시스템

2.3 시스템 보정

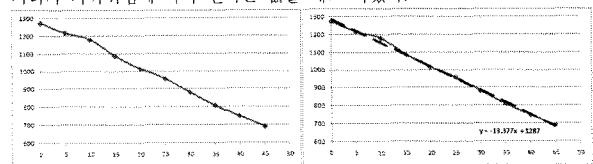
카메라에서 실제 마커의 위치에 따라서 바뀌는 x축 값과 y축 값에 대한 방향을 높이의 변화에 따라 ARToolkit에서 계산되어진 Marker의 변화를 측정하였다.



<그림 3> 실험 환경 구성

2.3.1 보정 실험

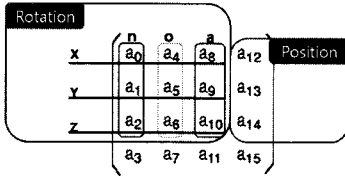
Marker와 Camera의 실 거리 측정을 위해 본 system에서 사용 중인 USB camera의 특성으로부터 나오는 값을 가지고 카메라 parameter값을 구하였다. 임의의 기준점을 잡고, 그 기준점으로부터 Marker와 카메라 간의 거리가 가까워짐에 따라 변하는 값을 체크 하였다.



(1) ARToolkit으로부터 나오는 값은 대략적이고 일변화된 값이기 때문에 실제 거리를 측정하기 위해선 현재 사용하고 있는 Camera의 Parameter값을 계산해주어야 한다. (1)부터 나온 데이터를 가지고 평균적인 선형 값을 구할 수 있다.(2).

2.4 이미지 획득

ARToolkit으로 부터 받은 이미지에 대한 정보 중 마커의 위치 정보를 통하여 x,y,z축 방향을 표시하는 이미지를 드로잉 할 수 있다. 마커의 위치에 대한 정보를 4 * 4 matrix로 받아 올 수 있다.



〈그림 4〉 matrix 구조

받아 온 matrix로 부터 회전은,

$$R(\psi, \theta, \phi) = R(z, \phi)R(y, \theta)R(x, \psi) \quad (1)$$

$$= \begin{bmatrix} c\phi c\theta & c\phi s\theta \psi - s\phi c\psi & c\phi s\theta \psi + s\phi c\psi \\ s\phi c\theta & s\phi s\theta \psi + c\phi c\psi & s\phi s\theta \psi - c\phi c\psi \\ -s\theta & c\theta \psi & c\theta \psi \end{bmatrix}$$

(1)로 나타낼 수 있다.

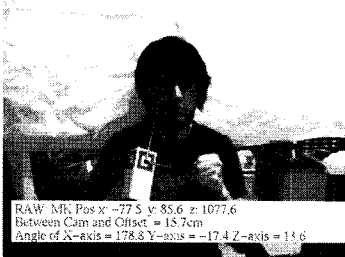
위의 식을 각 축에 대한 회전으로

$$R(z, \theta) = \begin{bmatrix} c\theta & -s\theta & 0 \\ s\theta & c\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2)$$

$$R(y, \phi) = \begin{bmatrix} c\phi & 0 & s\phi \\ 0 & 1 & 0 \\ -s\phi & 0 & c\phi \end{bmatrix} \quad (3)$$

$$R(x, \psi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\psi & -s\psi \\ 0 & s\psi & c\psi \end{bmatrix} \quad (4)$$

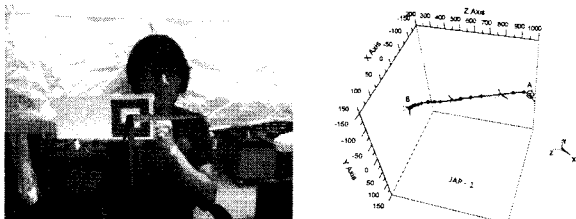
(2),(3),(4)으로 계산되어 진다.



〈그림 5〉 Marker 검출 및 3차원 좌표 그려주고, 위치, 축에 대한 회전, 거리를 표시

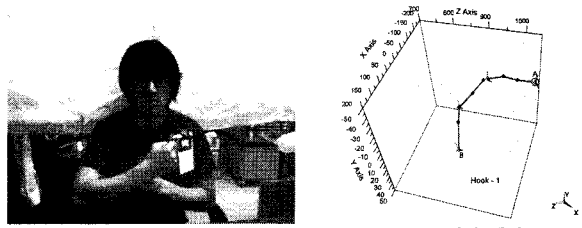
2.4.1 모션 데이터 획득

처음의 기본자세에서 팔을 뻗는 자세를 취하게 되면 복싱의 jab과 같은 동작이 된다. 이 동작의 움직이는 궤적을 추적해 해 보면, 그림 6과 같은 결과를 얻을 수 있다.



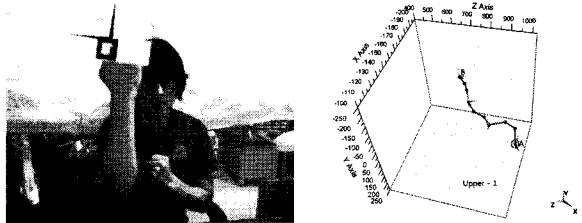
〈그림 6〉 펀치 jab의 마지막 이미지와 프레임 궤적

복싱에서의 hook 펀치의 경우, x축에 해당하는 운동을 진행한다. 하지만 움직인 궤적을 보면 기본자세로부터 시작된 운동은 그림 7과 같은 궤적으로 나타난다.



〈그림 7〉 펀치 hook의 마지막 이미지와 프레임 궤적

상대박의 턱을 가격하는 복싱의 기술인 uppercut는 대부분을 y축에 대한 운동으로 해석할 수 있다.



〈그림 8〉 펀치 uppercut의 마지막 이미지와 프레임 궤적

3. 결론 및 향후 연구 방향

본 논문에서 USB camera와 Marker 간의 실 거리 측정을 위한 실험과 Marker의 Matrix를 통하여 x, y, z축의 회전을 구할 수 있었다. 또한 추가 보정을 위한 Marker를 90°에 위치함으로써 복싱에서 일어나는 모션에서 모두 Detecting이 가능하도록 하였다. 프레임 단위에서 발생하는 Marker의 움직임과 회전을 연구 하였다. 하지만 기존 마커의 보정을 위해 90°에 위치한 Marker에 대한 추가적인 오차 수정에 필요하고 향후, 모션에 대한 이벤트가 발생하였을 때 application에서 이를 받아 모션을 표현할 수 있도록 하여, 실제 가정용 게임에 적합한 System을 개발 할 수 있도록 한다.

[참 고 문 헌]

- [1] <http://www.nintendo.co.kr/Wii/>
- [2] <http://www.hitl.washington.edu/artoolkit/>
- [3] <http://www.openscenegraph.org/>
- [4] Saeed B. Niku, "Introduction to ROBOTICS, ANALYSIS, SYSTEM S, APPLICATIONS," 69 ~ 76, Prentice, 2002
- [5] David H. Eberly, "GAME PHYSICS," 534 ~ 546, ELSEVIER, 2003