

동작 비디오 분석을 통한 캐릭터 애니메이션 생성

임예슬, 박종승
 인천대학교 컴퓨터공학과
 imthree@inu.ac.kr, jong@inu.ac.kr

Character Animation Generation from Motion Video Analysis

Ye-Seul Im, Jong-Seung Park
 Dept. of Computer Engineering, Incheon National University

요약

본 논문에서는 동작 비디오를 분석하고 이를 3차원 캐릭터 애니메이션으로 생성하는 방법을 제시한다. 비디오에서의 사람의 동작 인식에서는 OpenPose를 사용하여 사람의 몸 keypoints를 추적해 2차원 위치 좌표로 얻는다. 3차원 캐릭터 애니메이션으로 생성하기 위해 Deep-Learning을 사용하여 2차원 위치 좌표를 3차원 위치 좌표로 변경한다. 캐릭터 스켈레톤에 적용하기 위해 3차원 위치 좌표를 회전값으로 변환하고 그 회전값을 캐릭터 스켈레톤 좌표에 맞게 변환한다. 비디오의 사람의 동작과 유사한 3차원 캐릭터 애니메이션을 생성하는 방법을 제안하고 적용 결과를 제시한다.

1. 서론

3차원 캐릭터 애니메이션은 영화, 광고, 애니메이션, 게임, 영상 등 다양한 곳에서 사용되며 중요한 위치를 차지하게 되었다. 특히, 최근 가상 아이돌의 등장으로 3차원 캐릭터 애니메이션이 상업적 측면에서 다양한 매체들과 상호작용을 함으로써 더욱 발전되고 있다[1]. 오락 산업이 발전하고 캐릭터 애니메이션에 대한 수요가 증가하면서 새로운 애니메이션 제작 방법이 바로 모션 캡처(Motion Capture) 방식이다. 모션 캡처는 댄스 게임, FPS, 스포츠 게임 등 게임의 장르가 다양화되면서 기술의 활용도가 커지고 있다[2].

기존의 해당 분야에서 진행되었던 연구들은 Human Motion Capture Data(HMCD)를 통해 인간의 행동을 가장 잘 모방할 수 있는 휴머노이드 로봇에 인간의 행동을 동일하게 적용하는 연구, 비디오 감시, 환자 모니터링, 인간과 컴퓨터의 상호작용 등과 같은 많은 응용 프로그램들을 위한 지속적인 인간의 행동을 인식하는 연구 등 다양하게 발전되어 왔다[3][4].

OpenPose는 카메라의 이미지를 통해 사람의 뼈대를 추출하는 딥러닝 모델을 사용하여 사람의 얼굴, 신체 부위, 손가락 등의 keypoints를 실시간으로 추

적하여 좌표로 나타내는 API이다[5]. OpenPose를 사용하여 비디오 속 사람의 동작을 사람의 몸 keypoints로 추적하여 2차원 위치 좌표를 얻을 수 있는데, 이를 활용하여 다양한 연구를 수행할 수 있다. 교실에서 책상에 앉아 있는 학생의 자세를 keypoints로 추적하여 좋은 자세와 좋지 않은 자세를 확인하는 연구, 사람의 얼굴의 다양한 keypoints를 추적하여 사람의 기분을 추정하는 연구 등 OpenPose를 이용하여 다양한 분야의 연구들이 진행되고 있다[6][7].

본 논문은 동작 비디오를 통해 프레임마다 사람의 동작을 keypoints로 인식하고 이를 이용하여 캐릭터 애니메이션을 생성하는 방법을 제안한다. 2장에서는 비디오에서 사람의 동작을 OpenPose를 통해 2차원 위치 keypoints를 인식하고 이 2차원 위치 좌표를 Deep-Learning을 이용해 3차원 위치 좌표로 변경하여 회전값을 구하고 캐릭터 스켈레톤에 맞는 좌표계로 변환한다. 그 회전값을 캐릭터 스켈레톤 로컬 좌표로 만들어 각각의 Skeleton Joint에 적용하여 3차원 캐릭터 애니메이션을 생성하는 방법을 설명한다. 3장에서는 비디오에서 사람의 동작과 캐릭터 애니메이션을 비교하여 동작 비디오와 유사한 애니메이션이 생성되었다는 것을 보여준다. 마지막으로 4장에서 결론을 맺는다.

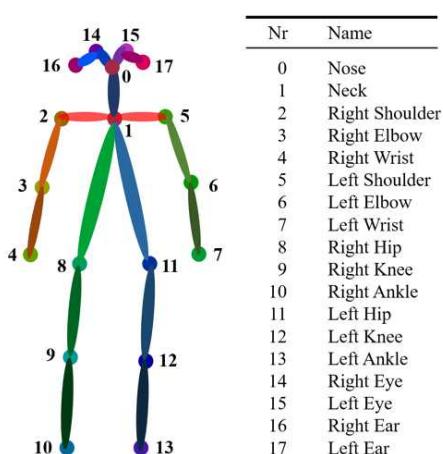
2. 3차원 캐릭터 애니메이션 생성

동작 비디오에서 3차원 캐릭터 애니메이션 생성을 위해 다음과 같은 순서로 진행된다. 동작 비디오를 OpenPose로 입력하여 2차원 keypoints 위치 좌표를 얻어낸다. 2차원 위치 좌표를 Deep-Learning을 사용하여 3차원 위치 좌표를 예측한 후 3차원 위치 좌표로부터 캐릭터 스켈레톤 각각의 회전값을 얻어낸다. 마지막으로 캐릭터 스켈레톤의 좌표에 맞게 회전값을 변환하여 3차원 캐릭터 애니메이션을 생성한다.

3차원 캐릭터 애니메이션을 생성할 때 캐릭터 스켈레톤에 적용하기 위해 각각의 과정에서 얻어낸 위치 좌표와 회전값을 캐릭터 좌표계로 변환해야 한다. 3차원 캐릭터 애니메이션을 생성하는 과정에서 Skeleton Joint들은 C_{World} (원점 기준 좌표계), C_{Pose} (root의 로컬 좌표계) 좌표계로 나타낼 수 있다. 3차원 캐릭터 애니메이션을 생성하기 위해서 각각의 Skeleton Joint 회전값을 C_{Pose} 에서 캐릭터의 기본 자세(rest pose) 기준의 Skeleton Joint 로컬 좌표로 변환 후 캐릭터 스켈레톤에 적용해야 한다.

2.1. 2차원 위치 좌표 계산

비디오를 OpenPose에 입력하면 매 프레임마다 사람의 동작 keypoints를 추적하여 눈, 귀, 코, 목, 어깨, 팔꿈치, 팔목, 엉덩이, 골반, 무릎, 발목 총 18개의 Skeleton Joint 각각의 좌표 $\{x, y\}$ 가 출력된다. (그림 1)은 OpenPose로부터 얻어낸 18개의 Skeleton Joint의 위치와 각각의 이름을 나타낸다.

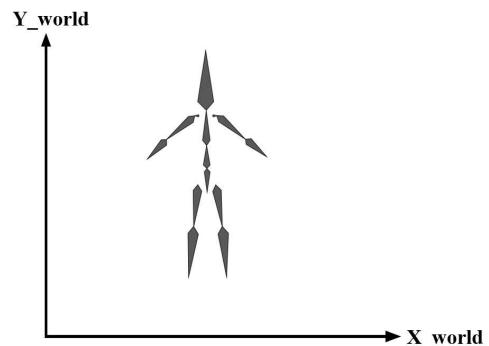


(그림 1) Skeleton Joint 위치와 이름

2.2. 3차원 위치 좌표 예측

OpenPose로부터 얻어낸 2차원 위치 좌표 $\{x, y\}$

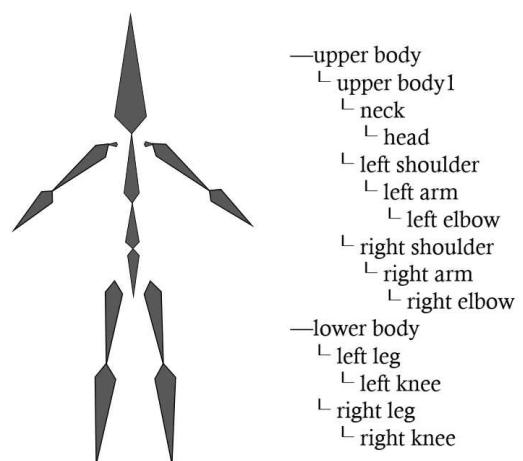
를 3차원 위치 좌표 $\{x, y, z\}$ 로 예측하기 위해 Deep-Learning을 사용한다[8]. Deep-Learning으로 얻어낸 3차원 위치 좌표 $\{x, y, z\}$ 는 원점을 기준으로 하는 World 좌표계 C_{World} 를 사용한다. (그림 2)는 C_{World} 에 존재하는 캐릭터의 Skeleton Joint를 보여준다.



(그림 2) World 좌표계에 있는 캐릭터 Skeleton Joint

2.3. 위치 좌표로부터 회전값 계산

OpenPose를 통해 동작 비디오에서 사람의 동작 keypoints를 추적해 Skeleton Joint의 위치 좌표 $\{x, y\}$ 를 얻어내고 Deep-Learning을 통해 3차원 위치 좌표 $\{x, y, z\}$ 를 예측하였다. 3차원 캐릭터 애니메이션 생성을 위해 3차원 위치 좌표를 회전값으로 계산하는 과정이 필요하다. 회전값을 구하기 위해서는 캐릭터의 Skeleton Joint의 계층 구조를 이해해야 한다. (그림 3)과 같이 캐릭터의 Skeleton Joint는 root와 부모, 자식이 존재하는데 이 계층 구조에 따라 회전값 계산이 달라진다.



(그림 3) Skeleton Joint 계층 구조

Skeleton Joint에서 root는 자기 자신의 회전값을 가지며, 부모를 가지는 Skeleton Joint들은 자기 자

신의 회전값에서 부모의 회전값의 역행렬을 곱해 단계적인 회전 변환이 필요하다. (그림 4)는 Skeleton Joint의 계층 구조에 따른 회전값 변환 방법을 나타낸다.

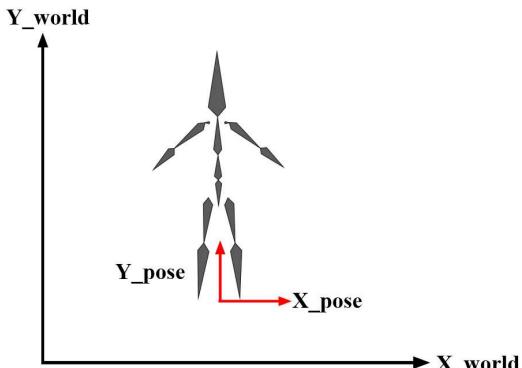
Algorithm 1 Convert from World Space To Pose Space

```

1: Initialize Skeleton Joint Quaternion Rotation  $\mathbf{R}$ .
2: Initialize Bone Skeleton Hierarchy  $\mathbf{H}$ .
3:
4: Input Skeleton Joint Quaternion Rotaton  $\mathbf{R}$ .
5: Input Bone Skeleton Hierarchy  $\mathbf{H}$ .
6: for  $b \in \text{Skeleton}$ 
7:   if  $b$  is a root
10:     $\bar{\mathbf{b}} \leftarrow \mathbf{R}_b$ 
11:   else
12:     $b\_parent \leftarrow H_{b\_parent}$ 
14:     $\bar{\mathbf{b}} \leftarrow R_b * \text{toParent}(R_{b\_parent})^{-1}$ 
15: return  $\bar{\mathbf{R}} = \{\bar{\mathbf{b}}\}$ ;
16:
17: function toParent( $R_b, \mathbf{H}$ )
18:   if(bone_name ==  $H_{root}$ )
19:     return  $R_b$ 
20:   else
21:     return  $R_b * \text{toParent}(R_{b\_parent})$ 
```

(그림 4) 계층 구조에 따른 회전값 계산

이를 통해 얻어낸 회전값은 Skeleton Joint 중 root를 기준으로 하는 로컬 좌표계 C_{Pose} 를 사용한다. (그림 5)에서 캐릭터 바로 아래에 있는 좌표계가 C_{Pose} 로, 캐릭터 Skeleton Joint의 root가 기준이 되기 때문에 C_{World} 좌표계의 어디에 위치하든 Skeleton Joint들의 변화가 없다면 동일한 값을 가지게 된다.



(그림 5) World 좌표계와 Pose 좌표계 비교

2.4. 캐릭터 스켈레톤에 회전값 적용

3차원 캐릭터 애니메이션을 만들기 위해 Skeleton Joint들의 회전값을 C_{World} 좌표계로부터 C_{Pose} 좌표계로 변환했다. Skeleton Joint들은 각각 자신의 로컬 좌표계를 사용하여 회전하기 때문에 3차원 캐릭터 애니메이션을 생성하기 위해서 C_{Pose}

좌표계를 사용하는 회전값은 기본 자세 기준의 로컬 좌표로의 변환이 필요하다.

기본 자세는 캐릭터 모델의 기본 포즈로 캐릭터 모델마다 각각 다른 값을 가지며 각 Skeleton Joint마다 좌표 $\{x, y, z\}$ 하나만을 보유하고 있다. C_{Pose} 좌표계에서 기본 자세를 취하고 있는 각각의 Skeleton Joint들의 좌표를 Local Matrix라고 한다. Local Matrix는 캐릭터 모델마다 하나씩 존재하며, 다른 자세를 취하고 있더라도 동일한 값을 지닌다. 또한 캐릭터 모델이 달라질 때마다 값이 변한다.

(그림 6)은 C_{Pose} 좌표계에 있는 회전값을 Local Matrix를 사용하여 Skeleton Joint 각각의 로컬 좌표로 변환하는 과정을 나타낸다. 최종적으로 로컬 좌표 회전값을 각 Skeleton Joint에 적용하면 동작 비디오와 유사한 캐릭터 애니메이션이 생성된다.

Algorithm 2 Make Bone Local Rotation in Pose Space

Inputs: Skeleton Joint Quaternion Rotation \mathbf{R} and Character Rest Pose Matrix \mathbf{M} .

```

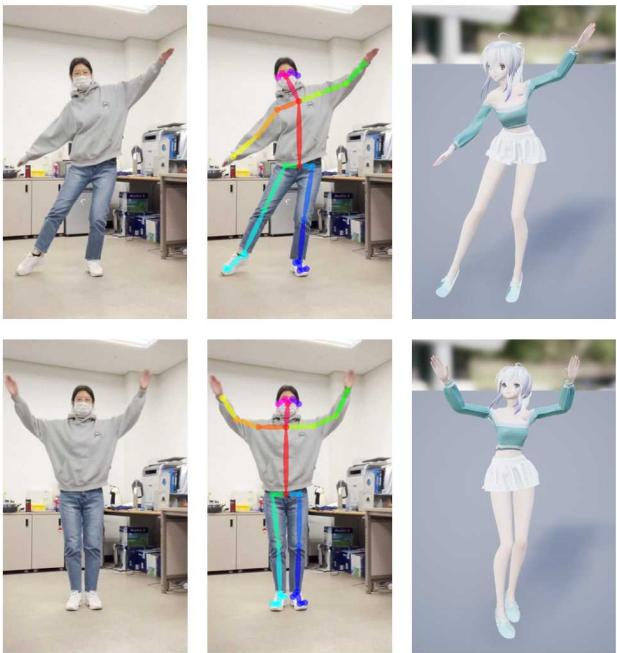
1: for  $b \in \text{Skeleton}$ 
2:    $s \leftarrow \sqrt{1 - R_b.w * R_b.w}$ 
3:   axis  $\leftarrow \{R_b.x/s, R_b.y/s, R_b.z/s\}$ 
4:   angle  $\leftarrow 2 * \arccos(R_b.w)$ 
5:    $\bar{\mathbf{b}} \leftarrow \text{Quaternion}(\mathbf{M} * \text{axis}^{-1}, \text{angle}).\text{normalized}$ 
6: return  $\bar{\mathbf{R}} = \{\bar{\mathbf{b}}\}$ ;
```

(그림 6) C_{Pose} 에서 로컬 좌표로의 회전값 변환

3. 실험 결과

본 논문에서 제안하는 방법은 동작 비디오에서 사람의 동작을 분석하여 같은 동작의 캐릭터 애니메이션을 만들기 위해 OpenPose와 Deep-Learning, 회전값 계산, 좌표계 변환을 사용한다. 본 논문의 실험은 Intel Core i5 CPU, NVIDIA Geforce GTX 970, 32GB RAM, Windows 10 64bit OS의 시스템에서 수행하였다. 실험 코드는 C++ 언어, Python 언어, OpenPose 라이브러리, Unreal Engine을 이용하여 구현하였다.

동작 비디오 중 두 개의 프레임을 사용하여 실험을 수행하였다. (그림 7)의 첫 번째 영상은 동작 비디오, 두 번째 영상은 OpenPose를 사용하여 Skeleton Joint들의 keypoints를 나타낸 영상, 세 번째 영상은 Skeleton Joint들의 2차원 위치를 3차원 위치로 변환하고 캐릭터 모델의 스켈레톤 계층 구조를 이용해 계산한 회전값과 그 값을 Skeleton Joint 로컬 좌표로 변환하여 캐릭터 스켈레톤에 적용해 생성한 3차원 캐릭터 애니메이션 영상이다.



(그림 7) 동작 비디오 분석을 통해 생성한
3차원 캐릭터 애니메이션

4. 결론

본 논문에서는 동작 비디오를 분석하여 캐릭터 애니메이션을 생성하는 방법을 제안하였다. 제안하는 방법은 실험을 통해 2차원 위치 좌표로부터 3차원 위치 좌표를 얻어내고 스켈레톤 계층 구조에 따른 회전값을 계산하여 캐릭터 스켈레톤에 적용 가능하도록 로컬 좌표로 변환해 캐릭터 애니메이션을 생성할 수 있음을 확인하였다. 제안 방법을 통하여 동작 비디오를 분석하고 이를 캐릭터 애니메이션으로 제작할 수 있다는 점에서 기여점이 있다.

향후 연구 과제로 비디오에서의 사람의 동작과 생성한 3차원 캐릭터 애니메이션이 동일한 행동을 취하기 위해 2차원에서 3차원 좌표로 예측하는 값의 오차를 줄이기 위한 연구가 필요하다. OpenPose에서 얻어온 2차원 좌표를 Deep-Learning을 통해 3차원 좌표로 예측한 것이기 때문에 비디오에서의 사람의 동작과 동일한 좌표를 예측할 수 없다는 한계점이 있다. 동일한 좌표를 가지고 회전값을 계산한 것이 아니기 때문에 비디오에서의 사람의 동작과 생성한 3차원 캐릭터 애니메이션이 유사하지만 동일하다고는 볼 수 없다. 또한 사용하고자 하는 모델의 Local Matrix를 직접 계산해 주어야 한다는 점에서 실시간 처리가 불가능하므로 이를 최적화하고 가속화할 수 있는 후속 연구가 필요하다.

Acknowledgments

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (No. NRF-2019R1F1A1060828).

참고문헌

- [1] 백승만, “3차원 캐릭터 애니메이션 산업현황에 관한 연구”, Journal of Korean Society of Design Science, Vol. 14, No.2, pp. 87–94, 2001.
- [2] 유석호, 박용현, 경병표, 이동열, 이완복, “모션캡쳐 데이터 활용을 위한 3D 게임캐릭터애니메이션 제작파이프라인”, 한국콘텐츠학회논문지, Vol. 8, No. 7, pp. 120–127, 2008.
- [3] D. Anirvan, N. Yoshihiko, “Making Feasible Walking Motion of Humanoid Robots From Human Motion Capture Data”, IEEE International Conference on Robotics and Automation, pp. 1044–1049, 1999.
- [4] B. Mathieu, B. Saïda, B. Boubakeur, G. Erwan, “Ongoing human action recognition with motion capture”, Pattern Recognition, Vol. 47, No. 1, pp. 238–247, 2014.
- [5] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, Y. Sheikh, “OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields”, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 43, No. 1, pp. 172–186, 2019.
- [6] C. Kehan “Sitting Posture Recognition Based on OpenPose”, IOP Conf. Series: Materials Science and Engineering, Vol. 677, No. 3, 2019.
- [7] Y. Zhengyuan, Z. Yixuan, L. Jiebo, “Human-Centered Emotion Recognition In Animated GIFs”, IEEE International Conference on Multimedia and Expo, pp. 1090–1095, 2019.
- [8] M. Julieta, H. Rayat, R. Javier, J. James, “A simple yet effective baseline for 3D human pose estimation”, IEEE International Conference on Computer Vision, pp. 2640–2649, 2017.