

A Real-time Motion Adaptation Method using Spatial Relationships between a Virtual Character and Its Surrounding Environment

Dongsik Jo*, Myung Geol Choi**

Abstract

Recently, character motion have been used extensively in the entertainment business, and researchers have investigated algorithms of reproducing, editing, and simulating mimic human movements. Also, many recent researches have suggested how a character interacts with its surrounding environment in terms of motion. Specially, spatial relationships of the environment have been introduced for adapting and preserving character motion. In this paper, we propose a motion adaptation technique preserving a spatial property between a virtual character and the configuration of its surrounding space. Additionally, we report on experimental results of smoothly adapted motions in various environmental structures with original motions such as walk, jump, and tumbling.

▶Keyword: Character motion, Spatial relationship, Motion adaptation, Motion capture

I. Introduction

최근, VR/AR, 게임/엔터테인먼트 등 다양한 분야에서 캐릭터 모션 애니메이션 기술이 활발히 적용이 되고 있다. 이러한 캐릭터의 모션 애니메이션을 제작하는 과정은 주로 모션 캡처 시스템을 이용하여 생성하거나 혹은 key-frame 애니메이션을 통해 만들어 지고 있다[1-2]. 이렇게 제작된 캐릭터 모션 애니메이션은 새로운 환경(예. 지형 혹은 물체 등)에 맞게 변형될 필요가 있을 것이다. 예를 들면, 평평한 평지를 걷는 동작을 하는 캐릭터의 애니메이션 동작은 지형의 높낮이가 변화하였을 경우 그 지형에 맞도록 새로이 캐릭터 모션을 수정하여 제작해야 한다. 기존 게임 콘텐츠 등에서 모션을 적용하는 경우 캐릭터 모션은 groundCheck를 통한 충돌 처리(collision detection)을 적용하여 캐릭터 동작을 지형에 맞게 업데이트하는 방식으로 운용하고 있고, 기존의 캐릭터 모션 애니메이션과 관련된 연구에서는 초기부터 설정된 환경(순간적으로 변화가 일어나지 않는 환경)에서 동작하는 캐릭터 모션을 표현하거나 변형하는 연구에 집중하고 있다[3]. 하지만, 캐릭터의 모션을 새로운 환경에 상호작용하는 경우와 사용자가 환경을 수정하는 경우에는 캐릭터의 원본 모션 데이터를 가지고 spatial

relationship을 유지하면서 관절 단위로 수정하여 변형하는 적응형 모션 자동 편집 방법이 필요할 것이다[3]. 이에 본 논문에서는 실시간으로 변화하는 지형 환경에서 캐릭터의 관절 위치를 계산하여 인체 동작을 적응적으로 표현하고, 이를 통해 캐릭터와 환경 사이의 연속적인 상호작용 모션을 자동으로 생성하기 위한 방법을 제시한다. 이를 위해 캐릭터와 상호작용하는 환경에서 공간적인 상관관계 정보를 포함하는 특징값을 추출하고, 추출된 특징 값을 기반으로 캐릭터가 실시간으로 환경에 적응하여 빠르게 모션을 변형하여 표현할 수 있도록 하는 방법을 제시한다. 또한, 캐릭터 주변 환경에 일치하고 자연스러운 모션 동작을 생성하기 위해 캐릭터의 신체 사이즈 제약 조건을 최대한 보전하도록 표현하는 방법을 제시하고자 한다.

요약하자면, 본 논문은 지형과 같은 주어진 환경을 기반으로 캐릭터 모션에 영향을 미치는 공간에 대한 특징값을 추출하고, 실시간으로 지형의 변화가 가능하도록 하고, 변화하는 환경에 캐릭터가 상호작용 하기 위해 환경 적응형 자동 모션 생성 방법을 제시한다. 특히, 다양한 지형 또는 지물 위에서 동작하는 걷는 동작, 점프 동작, 덤블링 동작 등을 이용하여 캐릭터 모션 정합 결과를 제시한다.

• First Author: Dongsik Jo, Corresponding Author: Myung Geol Choi

*Dongsik Jo (dongsik1005@wku.ac.kr), Professor, Dept. of Digital Contents Engineering, Wonkwang Univ.

**Myung Geol Choi (mgchoi@catholic.ac.kr), Professor, Dept. of Media Technology and Content Technology, Catholic Univ.

• Received: 2019. 01. 11, Revised: 2019. 03. 05, Accepted: 2019. 04. 01.

II. Related Works

본 논문에서 제시하고자 하는 캐릭터의 모션이 상호작용하는 환경에 적응하여 변형되는 몇 가지 연구 사례 및 연구 결과를 살펴볼 것이다. 대표적인 연구 결과 중 하나인 Lyard and Thalmman의 연구에서는 캐릭터가 지형의 self-penetration (지형의 안쪽으로 캐릭터의 발이 파고들어가는 현상)을 최소화하고, motion balance (캐릭터의 몸의 균형을 맞추기 위한 요소)를 맞추기 위한 모션 조정 방법을 제시하였다[4]. Ho et al. 의 연구에서는 두 명의 캐릭터 간 혹은 캐릭터와 상호작용하는 물체 간 모션에 영향을 주는 spatial relationship(공간적인 상관관계)을 정의하고, 이에 따라 자동으로 모션을 에디팅할 수 있는 수학적 모델을 제시하였다 [5]. 또한, Jo et al.의 연구에서는 depth camera를 기반으로 실시간으로 캡처되는 사용자의 skeleton joints의 위치 정보를 기반으로 하여 가상의 캐릭터가 상호작용 하는 물체(예. 가상의 의자)의 높이 변화에 적응적으로 모션이 적용되는 기술을 소개하였다[3]. 또한, Al-Asqhar et al.의 연구에서는 실시간으로 변형되는 지형의 정보에 반응하여 캐릭터의 관절의 위치 정보를 계산하고, 이를 이용하여 모션을 표현하는 기술을 소개한 바 있다[6]. 하지만, 기존의 연구들에 따르면 캐릭터의 모션의 오류를 최소화하기 위해 캐릭터 모션을 사용자 주변 환경에 맞게 최적화하는 것에는 한계가 있다고 할 수 있다. 따라서, 본 논문에서는 지형지물의 변화에 따라 모션에 영향을 줄 수 있는 특징 값을 추출하고, 환경과 캐릭터 사이의 정교한 상호작용 모션을 생성하는 기술을 제시하고자 한다.

III. Motion Adaptation Method

그림 1은 본 논문에서의 지형 환경 적응형 캐릭터 모션을 위한 방법을 제시하고 있다. 그림 1의 왼쪽 그림에서와 같이 서 있는 캐릭터의 모션은 오른쪽과 같이 지형의 높낮이가(그림에서 중간에 올라간 블록으로 표시된 영역으로 높낮이의 변화가 있음) 변화된 경우 이러한 환경적인 변화 요인을 반영하여 자동적으로 모션을 변경할 필요가 있겠다.

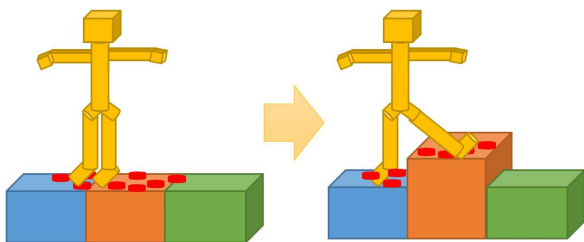


Fig. 1. Different configuration of the character's surrounding environment (e.g. a changed terrain in the center area), and an adapted motion to the different condition

표 1은 본 논문에서 제시한 실시간으로 변형되는 환경에 대해

자동으로 캐릭터 모션이 적응적으로 변형되는 과정을 알고리즘으로 설명하고 있다. 이를 구현하기 위해 논문에서 사용한 주어진 데이터는 환경에서는 특징값을 추출한 결과를 이용하였고, 캐릭터 모션을 수행하기 위해 걷기, 점프, 덤블링 3가지 정해진 모션 데이터를 입력으로 사용하였다. 그리고, 표 1의 알고리즘을 통해서 출력되는 결과는 캐릭터가 실시간으로 변화하는 환경에 적응하여 자동으로 최적화된 관절의 위치를 계산한 결과이다.

Table 1. Algorithm: our real-time motion adaptation method using spatial relationships

Iterative motion adaptation procedure
Input : Input motion Output: Adapted motion $D_{\text{descriptor_points}} \leftarrow D_{\text{environment}} + D_{\text{navigation_path}}$ //Initialization : Estimate relationship descriptor points in the given environment Repeat { for (i=1 to N){ //Calculate with all skeleton joints $p_{\text{estimate}} \leftarrow \sum (W_{\text{descriptor_point}} * D_{\text{descriptor_point}})$ //Estimate joint positions with all updated descriptor points by user's interaction ($W_{\text{descriptor_point}}$: weight among descriptor points) $p_{\text{update}} \leftarrow d * (p_{\text{current}} - p_{\text{previous}}) + 1/2 * (p_{\text{estimate}} - p_{\text{current}}) * 1/k$ //Estimate updated positions to apply smoothly adapted motion (d: damping coefficients, k: constant value) $p_{\text{new}} \leftarrow p_{\text{current}} + p_{\text{update}}$ //Approach to the desired joint position } $L_{\text{bone_length}} \leftarrow \sum (L_{\text{distance_between_joints}})$ //Estimate bone length with distance by all joints if ($L_{\text{bone_length}} < \text{threshold}$) then $p_{\text{correcting_new}} \leftarrow p_{\text{new}} + \Delta p_{\text{length}}$ //Correcting length between connected joints break; } } return $p_{\text{correcting_new}}$ //Result of joint positions for adapted motion

먼저, 주어진 환경 모델을 기반으로 모션에 영향을 주는 특징값 추출 과정을 수행하였다. 그림 1에서 지형 위에 작은 구형대로 표시된 점들은 높낮이에 관한 특징값 추출을 표현하였다. 환경에 대한 정확한 캐릭터 모션을 제어하기 위해 높낮이가 다른 다양한 바닥 지형 환경을 제시하고(혹은 실시간으로 높낮이를 사용자가 조정하도록 함), 이러한 지형의 공간적인 상관관계를 기반으로 다양한 캐릭터의 모션(예. 평평한 바닥에 걷기, 뛰기, 덤블링 등의 모션)이 자동적으로 변화된 지형에 적응되도록 리타겟팅 하도록 하였다.

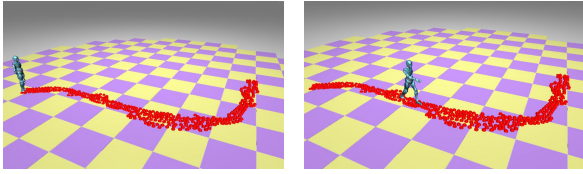


Fig. 2. Relationship descriptor points for the adapted character motion(spheres in the terrain)

그림 2의 지형 위에 작은 구로 표현된 점들은 본 논문의 실험에서 사용한 환경(지형) 모델에서 공간적인 상관 관계 정보를 포함하는 특징값을 계산하기 위한 초기 relationship descriptors(상관 관계 디스크립터)를 표현하였다. 그리고, 그림 3은 초기 relationship descriptors가 사용자에게 의해 높낮이가 변형된 지형에 맞게 수정된 결과이다. 여기에서 relationship descriptors는 모션 데이터에서 나타나는 캐릭터의 관절의 3차원 위치를 주변 환경모델에 대한 상대적인 위치로 계산하기 위해 특징 값을 표현하는 방법이다[6]. 예를 들어, 모션 데이터에 특정 프레임에서 나타난 j 번째 관절의 위치를 하나의 relationship descriptors를 사용하여 표현한다면 다음과 같은 식으로 표현될 수 있다.

$$P_j = d_i + \alpha_{i,j}n_i + \beta_{i,j}t_i + \gamma_{i,j}b_i \quad (1)$$

여기에서 P_j 는 캐릭터 skeleton joints들에서 j 번째 관절의 위치이고, d_i 는 i 번째 relationship descriptor points에 해당하는 지역 좌표계의 원점의 위치이다. 원점의 위치는 전역 좌표계로 표현되는 3차원 벡터이다. 그리고, n_i , t_i , b_i 는 지역 좌표계의 세 축으로 각각 relationship descriptor points에서의 환경 모델 표면에 대한 법선벡터(normal vector), 탄젠트벡터(tangent vector), 이항벡터(binormal vector)와 같다.

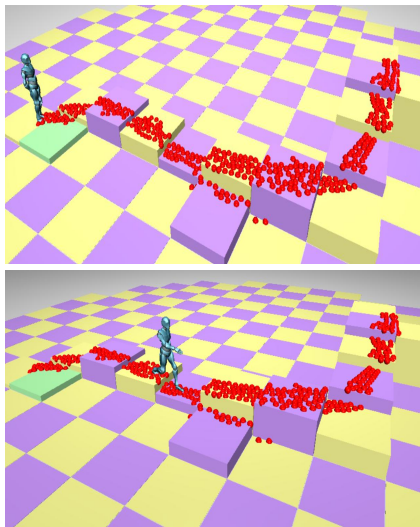


Fig. 3. Relationship descriptor points modified according to the change of terrain position(spheres in the terrain)

이 때, descriptor point d_i 는 P_j 에 가까운 곳에서 샘플링 되어야 하는데 환경 모델의 여러 부분 중 상호작용에 직접적으로 영향을 주는 것은 주로 주변 환경의 정보를 기반으로 하여야 하기 때문이다[6]. 그리고, 여러 주변 환경을 동시에 고려하면서 동시에 움직이는 P_j 에 대한 연속성을 유지하기 위해서는 서로 다른 위치에서 정의된 여러 개(최소 10개 이상)의 상관 관계 디스크립터를 사용하는 것이 필요하므로 여러 개의 디스크립터를 사용하여 캐릭터의 관절의 위치를 예측하는 식은 아래와 같이 표현될 수 있겠다.

$$P_j = \sum_{i=1}^N w_{i,j} (d_i + \alpha_{i,j}n_i + \beta_{i,j}t_i + \gamma_{i,j}b_i) \quad (2)$$

여기에서 $w_{i,j}$ 는 i 번째 디스크립터의 가중치이고, 가중치는 디스크립터 포인터 d_i 와 관절의 위치 P_j 사이의 거리에 반비례 되도록 정의하고, 일정 거리 이상이 되면 0이 되도록 하였다. Descriptor points를 샘플링하는 과정에서는 휴머노이드 각 관절 점에서 가까운 환경 모델의 표면위로 수직으로 프로젝션 한 지점으로 위치를 결정하였고, 밀도는 관절 위치와 주변 환경 모델 사이 거리에 반비례 하도록 샘플링하였다. 이렇게 샘플링 과정을 수행하면 캐릭터의 주변 환경에 근접하여 움직일 때 환경의 세부 모양 변화까지도 움직임에 영향을 줄 수 있도록 할 수가 있다.

다음 단계는 descriptor points들을 가지고 예측된 캐릭터 관절의 위치와 이전에 frame에서 주어진 캐릭터의 위치를 고려하여 새로운 관절의 위치를 찾는 것이다. 이는 캐릭터의 관절의 위치가 급격히 변화하는 것이 아니라 점진적으로 변화하도록 만드는 과정이다. 이를 위한 수식은 아래와 같다.

$$p_j^{update} = p_j^{current} + d * (p_j^{current} - p_j^{pre}) + 1/2 * (p_j^{estimate} - p_j^{current}) * 1/k \quad (3)$$

여기에서 d 는 damping coefficient이고, k 는 반복하려는 횟수 상수 값이다. 본 논문에서는 k 를 초기에 9로 설정하였다.

마지막 단계는 캐릭터의 전체 skeleton 크기의 범위를 고려하여 캐릭터 모션의 상관 관계를 보존하는 리타겟팅 방법을 적용하였다[7]. 즉, 환경 모델 모양이 변형될 때(예. 지형의 높낮이가 변화할 때) 모든 descriptor points에도 동일한 변형이 가해져 해당하는 지역 좌표계의 위치와 방향이 동시에 변화하게 된다. 수식 (1),(2),(3)에서 계산된 새로운 descriptors 위치와 좌표축을 기반으로 복원된 환경 모델에 대한 새로운 관절 위치 값을 계산할 수가 있지만 캐릭터의 인체에 계산된 관절의 위치 값을 직접으로 대입할 수가 없다. 왜냐하면, 캐릭터의 관절 위치를 계산할 때 캐릭터의 skeleton 각 관절 사이의 거리가 항상 일정해야 한다는 제약 조건을 만족해야 하기 때문이다. 이를 위해서는 아래 수식과 같이 관절 값 간의 연결된 길이를 보존하는 반복 계산 방법을 이용하였다.

$$\Delta p_j = \frac{p_j - p_k}{\|p_j - p_k\|} (l - \|p_j - p_k\|) \quad (4)$$

여기에서 p_k 는 j 번째 관절(joint)과 연결된 상위 관절의 위치이고, l 은 두 관절 간의 길이이다. 본 연구에서는 연속적인 캐릭터의 모션 동작(걷기, 점프, 덩블링)을 가지고, 원본 모션 데이터를 시간에 따라 변하는 정지 자세의 연속적인 집합으로 표현하였고, 환경에 적응된 최적화된 캐릭터 모션을 계산하였다.

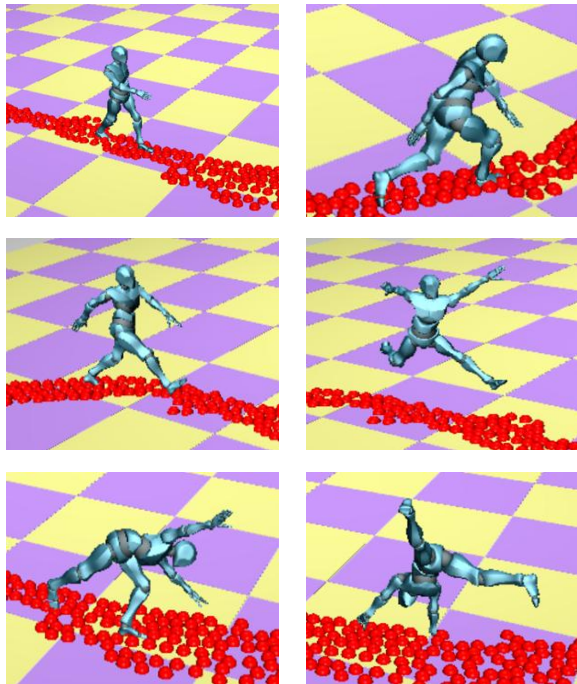


Fig. 4. Original motions of the character(e.g.walk, jump, tumbling) and descriptor points placed in the path

이러한 모션 최적화 문제를 풀기위해서 멀티 해상도 모션에 디팅 기법을 사용하였다[7]. 즉, 실시간으로 변화되는 관절의 위치를 계산하기 위해서는 일정한 시간 간격으로 샘플링하여 IK(Inverse Kinematics)를 적용한 후, 각 구간 사이는 스플라인 보간 (Spline Interpolation)을 적용하여 전체 모션을 완성한다. 그리고, 오차가 주어진 임계치 이하로 떨어질 때까지 샘플링 해상도를 높여가면서 이 과정을 반복하였다(Table 1 참조). 이 방법은 전체 시간 구간을 한 번에 최적화하는 방법에 비해 계산이 빠르고 최종 동작의 연속성이 보장된다는 장점이 있다. 참고로, 본 논문에서 사용된 캐릭터 모션 데이터 적용에서는 head, neck, root, left/right hip, left/right knee, left/right ankle, left/right foot, left/right toe 로 skeleton이 구성된 휴머노이드형 인체 모델을 반영한 캐릭터를 이용하였다.

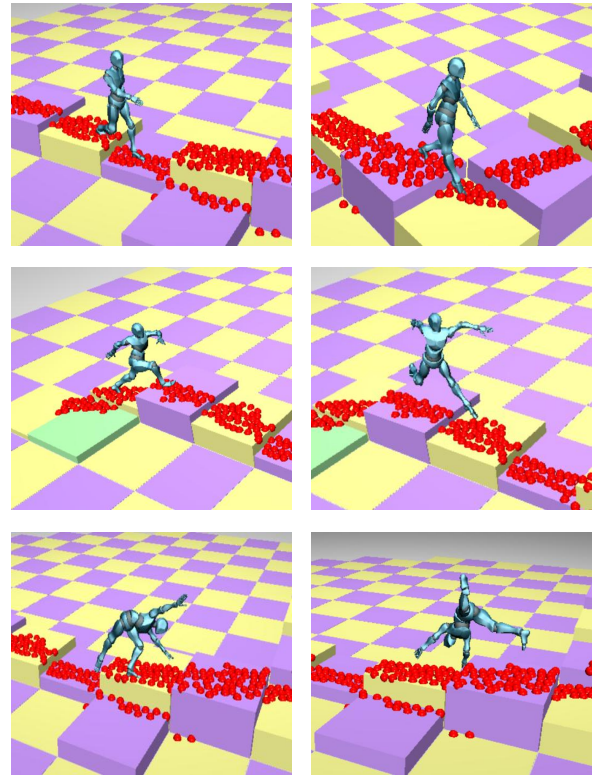


Fig. 5. Adapted motions with spatial relationships of the character's surrounding environment(e.g.walk, jump, tumbling motions)

IV. Experiment

본 연구에서는 다양한 형태의 이동 동작을 대상으로 가상의 캐릭터가 환경 내에서 자유롭게 동작하는 것을 구동하는지를 평가하였다. 이미 언급한 바와 같이 사용한 데이터는 환경에서 특징값을 추출하도록 하였고, 캐릭터 모션을 수행하기 위해 걷기, 점프, 덩블링 3가지 정해진 모션 데이터를 사용하였다. 그림 4는 걷기, 점프하기, 덩블링 동작에 대해서 단순히 지정된 trajectory를 가지고 모션 모델을 적용한 기초 데이터의 결과를 보여주고 있다. 그림에서 표현된 작은 구로 표현된 점은 캐릭터 모션과 바닥환경 사이의 상관 관계를 추출하기 위해 계산된 descriptor points를 나타낸다. 그림 5는 캐릭터가 이동하는 도중에 지형에 대한 바닥의 높이를 변형시켜 적응형으로 동작하는 모션 데이터를 계산한 결과이다. 새로운 높이에 따라 descriptor points의 높이 변화가 반영되고, 이러한 업데이트를 반영한 descriptor points를 가지고, 자동으로 캐릭터의 자세를 추정하였다. 걷기, 점프, 덩블링 3가지 캐릭터 동작에 대하여 주어진 path에 대한 애니메이션을 수행할 때 실시간으로 지형을 사용자가 변화 가능하도록 하였고, 이에 대해 실시간으로 캐릭터의 모션을 환경에 맞게 리타겟팅한 결과 관절 충돌 결과(특히, 손 혹은 발)에 대한 물리적 오류없이 자연스러운 모션이

나타남을 확인할 수가 있었다. 즉, 주어진 캐릭터의 모션 데이터가 새로운 환경에 대해 적응하여 동작하도록 모션을 생성하는 것이 가능하였다. 또한, 주어진 적응형 모션을 만드는 방법은 30fps 이상 실시간으로 캐릭터의 모션이 동작 애니메이션이 표현이 됨을 알 수가 있었다. 알고리즘 구현 및 실험에 사용한 컴퓨팅 자원으로는 Intel Core i7-8700 CPU 3.20GHz, 32.0GB 메모리, windows 10 Pro 64비트 운영체제 환경에서 구동하였다. 표 2는 실시간으로 변화하는 지형에 따라 적응형 모션을 생성하는 캐릭터에 가지고 충돌 횟수와 에러율을 측정 한 실험 결과이다. 여기에서 충돌 횟수는 캐릭터의 다리 (덤블링의 경우는 팔과 다리)에 cube 형태의 oriented bounding box(OBB)를 가지고 지형과 충돌 횟수를 측정하였고, 에러율은 최초로 설정된 bone strength의 길이가 적응형 모션을 적용하여 변화된 정도를 가지고 측정하였다. 그리고, 지형의 변화된 정도는 캐릭터의 신체 크기를 고려하여 상(허리높이), 중(무릎 높이), 하(발목 높이)로 나누어서 수행하였고, 지형의 변화는 지그재그로 변화하도록 조절하였다 (첫 번째 칸이 높이면 두 번째 칸은 평평하게 두고, 세 번째 칸을 높이고, 네 번째 칸은 평평하게 두는 형식).

Table 2. Results of motion adaptation between the virtual character and its surrounding environment with the change of terrain position

Motion	Change of terrain position	Collision counts (Number)	Error rate (%)
Walk	High	5	4.6
	Medium	2	3.9
	Low	0	2.3
Jump	High	3	3.2
	Medium	2	2.9
	Low	0	2.6
Tumbling	High	7	5.3
	Medium	3	3.2
	Low	2	1.7

V. Results and Discussions

본 연구에서 환경에 대한 특징값 추출을 이용하고, 환경에 적응적인 캐릭터 모션에 대한 방법을 적용하였다. 이 방법은 추출된 특징 값으로 환경에 대한 정보를 설정하므로 지형의 변화에 따라 캐릭터 모션의 형태가 자동적으로 spatial relationship을 유지하면서 변화하는 것을 확인하였다. 즉, 사용자가 지형의 변화를 즉각적으로 변화하여도 자동으로 계산되는 특징 값들로 인해 캐릭터 모션이 즉각 반응하여 동작하는 것을 확인할 수가 있었다. 표 1의 결과에서는 cube OBB로 충돌 체크를 한 결과 각기 다른 지형의 변화에 따라 모션을 적용한 결과 7회 이하로

적게 나타나는 것을 알 수가 있었고, 에러율도 높이 변화된 덤블링 동작을 제외하면 5% 이하가 되는 것을 확인할 수가 있었다. 여기에서 충돌을 더 작게 만들려면 관절의 bone strength 뿐만 아니라 적응형으로 각각의 관절에 대한 회전 (특히, 지형과 충돌되는 부분)에 대한 부분도 고려하여 알고리즘을 개선해야 할 것으로 본다. 또한, bone strength 최적화를 조금 더 고속화된 컴퓨팅 시스템을 이용하여 찾을 수 있다면 길이에 따른 에러율도 줄일 수 있을 것으로 본다. 본 논문은 현재 VR환경과 같은 그래픽으로 표현된 공간에서 적용을 하였으나 제시된 방법을 이용한다면 연속적이고 복잡한 공간적 제약 조건이 있는 경우(예. 실제 환경을 캡처한 경우)에서도 적용이 가능할 것으로 본다.

본 논문에서 제시한 방법의 한계점은 캐릭터가 수행할 수 있는 모션의 한계를 넘을 경우(예를 들면, 걷는 동작을 적용할 경우 캐릭터의 키보다 더 높게 변화된 지형의 변화) 가능한 다른 모션으로 변형하여 적용할 수 있도록 해야 한다. 현재 구현된 방법은 환경에 대한 특징값을 획득하는 과정이 지형에 대한 위치만을 가지고 캐릭터의 모션을 적용하기 때문에 추후 연구에서는 환경과 캐릭터 모션 간의 상관 관계가 추출되어야 할 것이다. 예를 들면, 캐릭터가 점프할 수 있거나 없을 수 있거나 매달릴 수 있거나 등등 환경에서 캐릭터가 수행할 수 있는 행동 패턴을 자동으로 정의할 필요가 있을 것이다. 이를 위해서는 기계 학습 등 방법을 통해서 캐릭터의 모션이 자연스럽게 스위칭이 될 수 있도록 해야 할 것이다[8-9]. 즉, 기본적인 모션 동작을 계속 유지하는 것이 아니라 다양한 모션 데이터에서 변형된 환경에 자연스럽게 적응되는 모션을 찾고, 찾은 모션을 자연스럽게 적응형으로 변경할 수 있는 방법이 필요하다. 이는 다양한 환경 및 지형 데이터를 기반으로 캐릭터 모션을 학습하고, 생성된 지형에 대해 기존에 존재하던 모션 데이터를 무작위로 대입을 해보고 정합이 되는 경우 적용하고, 그 환경에 맞게 모션을 리타겟팅하는 방법을 적용한다면 변화하는 환경에 부합되는 모션 생성이 가능할 것으로 본다. 또한, 좀 더 세밀하게 캐릭터 주변 환경에 대한 물체와 상호작용 할 수 있도록 contact pose에 해당되는 적응형 캐릭터 모션을 만들고, 만들어진 모션과 주어진 환경과 충돌이 세밀하게 어느 정도 발생을 하였는지 평가를 할 필요가 있을 것이다[10-11].

VI. Conclusions

본 논문에서는 캐릭터의 주변 환경 지형지물에 따라 자동으로 original 모션 데이터가 최적화 및 리타겟팅되는 기술을 제시하였다. 변화된 공간에 적응하는 캐릭터 모션을 자동으로 생성하기 위해 공간(지형)에 대한 특징 값을 추출하고, 이를 이용하여 캐릭터 모션의 형태를 적응적으로 변화하는 알고리즘을 제시하였다. 그리고, 몇 가지 모션 적용을 통해 캐릭터의 bone

length 유지와 공간과의 충돌에 대한 영향 등을 가지고, 캐릭터 주변 환경과 캐릭터 간 공간적 상관 관계가 유지되는 것을 확인할 수 있었다. 본 논문에서 제시된 기술은 향후 캐릭터 애니메이션 제작 및 실시간 제어 기술, VR/AR/MR 환경에서의 캐릭터 제어 기술, 가상 캐릭터를 이용한 시각화된 콘텐츠 기술에 도움이 될 것으로 판단된다.

향후에는 본 논문에서 수행한 수직적인 바닥 환경에서 확장하여 실시간으로 실제 환경을 스캔하여 가정집, 사무실, 공공장소 등 다양한 환경에서 적용할 수 있는 혼합현실(MR) 캐릭터 모션에 적용을 할 계획이다. 즉, 보다 높은 복잡도를 가지는 실제 공간의 지형 혹은 장애물들을 회피하여 다닐 수 있도록 MR 캐릭터 모션을 구성하는 것에 적용하려고 한다. 또한, 본 논문을 기반으로 하여 적응형 캐릭터 모션 기초 연구를 확장하여 기계 학습을 이용한 캐릭터 이동 모션 연구를 진행 하려고 한다. 실제 환경과 같이 복잡한 공간적 제약조건이 있는 경우 계산량 증가와 즉각적인 캐릭터 모션 반응이 가능하도록, 캐릭터 자세 정보와 주변 지형지물 정보를 입력으로하여 캐릭터의 3차원 위치와 속도를 추정할 수 있는 알고리즘으로 구현을 확대하려고 한다.

REFERENCES

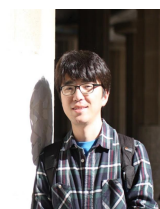
- [1] M. Choi, and K. Lee, "Interactive control of big-object manipulation animation," *Computer Animation and Virtual World (CASA 2016)*, Vol. 27, Issue 3-4, pp.435-442, 2016.
- [2] M. Choi, M. Kim, K. Hyun and J. Lee, "Deformable motion: Squeezing into cluttered environments," *Computer Graphics Forum*, Vol.30, No.2, pp.445-453, 2011.
- [3] D. Jo, K.-H. Kim, and G. Kim, "Avatar motion adaptation for AR based 3D tele-conference," *Proceedings of the IEEE Workshop on 3DCVE*, pp.1-4, 2014.
- [4] E. Lyard, and M. Thalmann, "Motion adaptation based on character shape," *Computer Animation and Virtual Worlds*, Vol.19, Issue 3-4, pp.189-198, 2008.
- [5] E. Ho, T. Komura, and C. Tai, "Spatial relationship pre-serving character motion adaptation," *Proceedings of SIGGRAPH ACM Transactions on Graphics*, Vol.29, No.4, Article 33, 2010.
- [6] R. Al-Asqhar, T. Komura, and M. Choi, "Relationship descriptors for interactive motion adaptation," *Proceedings of 12th ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pp.45-53, 2013.
- [7] J. Lee, and S. Shin, "A hierarchical approach to interactive motion editing for human-like figures," *Proceedings of the 26th annual conference on computer graphics and interactive technique*, pp.39-48, 1999.
- [8] D. Holden, T. Komura, and J. Saito, "Phase-functioned neural networks for character control," *ACM Transactions on Graphics (TOG)*, Vol.36, No.4, Article 42, 2017.
- [9] H. Jang, B. Kwon, M. Yu, S. Kim, and J. Kim, "A variational U-Net for motion retargeting," *Proceedings of SIGGRAPH Asia Posters*, Article No.1, 2018.
- [10] C. Kang, and S. Lee, "Environment-adaptive contact poses for virtual characters," *Computer Graphics Forum*, Vol.33, No.7, pp.1-10, 2014.
- [11] Z. Liu, A. Mucherino, L. Hoyet, and F. Multon, "Surface based motion retargeting by preserving spatial relationship," *Proceedings of the 11th Annual International Conference on Motion, Interaction, and Games(MIG)*, Article No.7. 2018.

Authors



Dongsik Jo received the Ph.D. degree in Computer Science and Engineering from Korea University, Korea, in 2017. He joined the virtual reality team, Electronics and Telecommunications Research Institute (ETRI) from 2004 and 2017, Daejeon, Korea.

Dr. Jo is currently an assistant professor in the Department of Digital Contents Engineering at Wonkwang University, Iksan, Korea. He is interested in virtual reality (VR), augmented reality (AR), collaborative virtual environment, virtual human, and HCI.



Myung Geol Choi received the Ph.D. degree in Computer Science and from Seoul National University, Korea, in 2010. After graduation, he worked as a postdoctoral researcher in University of Edinburgh, UK. Dr. Choi is currently an assistant professor

in the Department of Media Technology and Contents Technology at the Catholic University, Korea. He is interested in character animation, interactive animation authoring, human-computer interfaces, and all the technology that makes it easier for people to be creative.