# 복합적 행동들을 효율적으로 구현하기 위한 기본 동작의 재활용 기법

Reusing Technique of Primitive Motions for Effective Implementation of Complex Action

#### 최준성, 박종희

경북대학교 전자공학과

Jun-Seong Choi(daegulink@naver.net), Jong-Hee Park(jhpark@ee.knu.ac.kr)

#### 요약

가상 세계의 에이전트의 설계를 위해서 물리적 사실감뿐만 아니라 동적으로 변하는 상황에 대응하는 에이전트의 다양한 물리적 행동들의 구현이 필수적이다. 본 연구는 다양한 행동을 구현하기 위한 기본 동작들을 재활용한 복합적 행동 및 에이전트가 즉시 주위의 환경에 반응할 수 있는 기법을 구현한다. 또한 관절에 대한 에이전트의 신체 구조를 설계하고 기본 동작 또는 복합 동작을 실시간으로 표현한다. 이 기법을 구현하기 위해서, 기본적인 관절 회전을 위한 애니메이션 함수와 기본 동작으로 이루어진 전체적 동작을 구성하는 기능을 구현한다. 이러한 동작들은 에이전트의 목적을 달성하는 계획에 포함될 수 있다. 즉, 재활용된 기본 동작을 이용하여 다양한 행동을 쉽게 생성하는 기법을 개발한다. 본 연구는 '걸으면서 먹다'와 같은 병렬 행동을 구현할 때 뚜렷한 장점을 가진다. 이는 두 개 이상의 행동들은 우선순위에 따라 자연스럽게 병렬 행동으로 조합된다. 이를 증명하기 위해 몇 개의 복합 행동 및 병렬 행동을 구현한다.

■ 중심어: | 기본 동작 | 복합 행동 | 에이전트 | 동작 재활용 | 가상 세계 |

#### **Abstract**

Apart from the physical realism, the implementation of various physical actions of an agent to respond to dynamically changing situations is essential for the design of an agent in a cyber world. To achieve a maximum diversity in actions, we develop a mechanism that allows composite actions to be constructed by reusing a set of primitive motions and enables an agent to instantly react to changes in the ambient states. Specifically we model an agent's body in terms of joints, and a primitive or composite motion is performed in a real time. To implement this mechanism, we produce an animation for basic joint movements and develop a method to construct overall motions out of the primitive motions. These motions can be assembled into a plan by which an agent can achieve a goal. In this manner, diverse actions can be implemented without excessive efforts. This approach has conspicuous advantages when constructing a parallel action, e.g., eating while walking, that is, two or more parallel actions can be naturally merged into a parallel action according to their priority. We implement several composite and parallel actions to demonstrate the viability of our approach.

■ keyword: | Primitive Motion | Complex Action | Agent | Reuse of Motions | Virtual World |

\* 이 논문은 2013년도 정부(교육부)의 재원으로 BK21 플러스의 지원을 받아 수행하였습니다(21A20131600011).

\* 이 논문은 2013년도 경북대학교 학술연구비에 의하여 연구되었음.

접수일자 : 2014년 05월 29일 심사완료일 : 2014년 08월 18일

수정일자: 2014년 08월 18일 교신저자: 박종희, e-mail: jhpark@ee.knu.ac.kr

# 1. 서 론

가상 세계를 무대로 하는 응용 분야들에서 필요한 상황들을 구축할 때, 해당 응용분야의 성격에 따라 각 상황의 사실성(realism)이 중요시 될 수도 있고 상황들의 다양성이 우선시될 수도 있다[1]. 수많은 상황들을 제공하는 동시에 무한히 사실적으로 구현하는 것은 실제로는 불가능할 뿐 아니라 현실적으로 필요하지도 않다. 특히 개념적 측면이 중요한 응용분야들에서는 상황의다양성이 사실성(realism)보다 우선된다. 상황들은 관련된 객체들의 변화나 움직임을 통해 전개되어 나가는데 특히 다른 객체들에 비해 인간의 동작은 무수한 다양성을 가진다. 그에 상응한 가상 상황 내에서도 주체적으로 행위를 수행하는 에이전트들의 동작은 상황의다양성을 결정하는 핵심 요소가 된다.

3차원 게임을 위한 애니메이션과 휴머노이드 로봇 제 작 분야들에서 사람의 동작을 사실적으로 묘사하기 위 한 노력이 활발히 이루어지고 있다[2]. 이와 관련한 기 존의 연구들은 동작의 사실성에 중점을 두기 때문에 동 작의 다양성을 높이고 동작에 대해 주위 환경의 변화가 미치는 영향을 표현함에 있어 한계가 있을 수밖에 없 다. 따라서 구현할 수 있는 동작들의 개수를 대량으로 늘리는데 드는 막대한 비용에 비추어 다양한 동작들이 필수적인 시뮬레이션 분야에는 적합하지 않다. 한편 게 임에서 사용되는 캐릭터와 휴머노이드 로봇의 행동들 은 일반적으로 FSM(finite state machine)을 바탕으로 구현된다[3][4]. 즉, FSM은 미리 정의된 동작들의 종류 와 범위를 지정함으로써 에이전트의 상태를 쉽게 정의 할 수 있고, 전체적인 시스템을 간단히 구축할 수 있는 장점을 가지고 있다[3]. 그러나 FSM을 이용하여 구현 된 학습 시스템은 에이전트의 반응이 몇 개로 미리 고 정되어있기 때문에 학습자가 몇 번만 사용하게 되면 쉽 게 예측할 수 있게 되어 몰입감을 잃어버리기 쉽다. Linden Lab에서 개발한 가상 현실 게임인 'Second life' 는 기존의 게임과 달리 사용자가 만나는 모든 에이전트 를 실제 사람이 조종하고 사용자가 예측할 수 없는 수 많은 변수들을 발생시킨다. 이는 사용자의 몰입감을 유 지할 수 있게 만든다[5]. 그러나 'Second life'는 높은 자 유도가 장점이지만, 제한적인 객체에 한해서 동작을 생성할 수 있고 온라인 상태에서만 공유가 가능하다.

Electronic Art에서 개발한 게임인 'Sims'는 가상 세계 내에서 사용자가 인공지능인 'NPC(non player character)'와 상호 작용하고 인간의 기본적인 욕구들을 구현하고 적용함으로써 사용자가 다양한 활동을 하게 만든다[6]. 그러나 'Sims'에서 다루는 캐릭터들이 소수로 제한되어 있고 내용의 현실감이나 다양성이 부족하다. 따라서 현실감 있고 다양한 가상 상황들을 제공하기 위해서는 중요한 기본적 상위 개념들을 포함하고 해당 분야를 대표하는 확장 가능한 모델을 제공하는 온톨로지와 가상 세계 내에서 기본적인 요소인 역사적 문맥(historical context)을 형성하는 시간과 공간에 대한 지식을 체계적으로 저장하는 ST 그래프(spatio-temporal graph)의 통합적 상황표현모델이 필요하다[7][8].

본 논문은 가상세계 속에서 상황들의 핵심 동적 요소 인 에이전트의 다양한 동작들을 효율적으로 화면상에 시각적으로 나타낼 수 있는 구현기법을 개발한다. 이를 위해 먼저 기본 동작들을 재사용함으로써 수많은 동작 들을 효과적으로 구현하고 동시에 주변 상황의 변화에 실시간으로 대응할 수 있게 한다. 구체적으로 에이전트 의 신체를 몇 개의 관절들을 중심으로 구성하고, 그들 의 개별적 움직임을 병렬 실행함으로써 기본 동작 또는 복합 동작들을 실시간으로 생성할 수 있게 한다. 이를 구현하기 위해 각 관절의 기본적 움직임들을 애니메이 션으로 제작하고 신체 각 부분들을 조합하여 전체적인 동작들을 만들어내기 위한 방법을 개발한다. 이러한 동 작들은 관절의 독립적인 움직임들을 구성 및 조합하는 도구를 제공하고, 계획(planning)을 통해 유기적으로 조합하여 실행함으로써 에이전트가 정해진 목표를 달 성할 수 있게 한다. 그리고 예기치 못한 상황들에 즉시 대처하고 동시다발적으로 일어나는 여러 에이전트들의 행위를 효과적으로 구현하기 위하여 에이전트가 현재 실행하고 있는 행동을 실시간으로 다른 행동으로 전환 할 수 있는 방법을 고안한다.

#### 관련 연구

### 2.1 MPEG4(Moving Picture experts group 4)

MPEG4는 대역폭이 적은 통신매체에서도 전송이 가능하고 양방향 멀티미디어를 구현할 수 있는 audio/video 표준 부호 방식이다. MPEG4는 영상 내용에 근거하여 영상 신호를 부호화하는 방식을 추구하고 있고 기존의 방송, 가전, 컴퓨터, 통신 분야의 각 멀티미디어 정보를 디지털화함으로써 데이터를 융합할 수 있다[9]. MPEG4의 기능은 크게 객체 지향 대화형, 고능률 압축, 범용 액세스 등이다. 객체 지향 대화형 기능은화면이나 음향 같은 요소들을 독립적으로 취급하면서이들을 서로 링크에 의해 결합해 사용자가 화면이나 음향을 자유롭게 구성할 수 있다.

### 2.2 모션 캡처(Motion Capture)

모션 캡처는 3차원 공간에서 물체의 움직임을 측정하고 컴퓨터가 사용할 수 있는 형태의 정보로 기록해 분석 및 응용하는 기술이다. 모션 캡처가 게임, 영화, 광고등 다양한 분야에서 활용되는 이유는 사실적인 동작으로 표현이 가능하다는 것이다. 배우의 연기를 카메라로 촬영하여 동작을 수치화하고 캐릭터에 적용하면, 캐릭터는 배우의 동작을 실행하게 된다. 모션 캡처의 시스템은 실시간 동작을 촬영하는데 사용되는 몇 가지 기법이 있다. 모션 캡처의 각 기법을 구분하는 요인으로는데이터의 정확도, 샘플링(sampling) 속도, 배우에게 허용되는 동작의 자유 각도 등이 있다[10].

#### 2.3 시공간 그래프(ST Graph)

상황의 다양성을 높이기 위해서는 개별 에이전트의 행동에 관련된 시간적 공간적 배경을 체계적으로 고려하기 위한 통합적 지식구조가 필요하다. 이에 의해 부여되는 역사적 문맥(historical context)은 에이전트의 행동의 다양성을 여러 차원에서 한층 더 높이는 효과를 가져올 수 있다. 이를 위한 지식 표현 모델로서 ST 그래프 (spatio-temporal graph)가 개발되어있다[7]. ST 그래프는 다차원적이고 계층적인 형태로 상황의 시공간적 측면을 표현하게 된다[8].

### Ⅲ. ST 그래프를 사용한 상황 표현

## 3.1 계층적 구조 표현을 위한 좌표계

계층적 신체 구조는 각 신체 부분이 연결되는 부분과 상대적 관계를 가지기 때문에, 동작들은 모델 좌표계 (model coordinate system) 내에서 구현되고, 이러한 동작들은 외부 객체와 상호 작용할 필요가 없는 환경에 대해 효율적이다. 월드 좌표계(world coordinate system)는 외부 객체와 충돌 검출에 효율적이고 수 많 은 객체들을 포함하는 넓은 공간에 표현되는 상황에 대 해서 장점이 있지만, 자연스럽고 직관적인 움직임의 구 현에는 적합하지 않다[11]. 이러한 좌표계의 차이점은 직교 좌표계와 구면 좌표계로 비교할 수 있다.

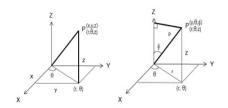


그림 1. 직교 좌표와 원주 좌표의 관계 및 구면 좌표와 원주 좌표의 관계

[그림 1]은 직교 좌표(x,y,z)와 원주 좌표(r,θ,z) 사이의 관계를 그림으로 나타낸 것이다. Z의 값은 두 좌표계에서 같은 값이고 (r,θ)는 평면(x,y)의 점(x,y)을 얻기위한 극 좌표의 순서쌍이다. 그림과 같이 직교 좌표와극 좌표의 관계에 따르면 다음 식을 얻는다.

$$x = rcos\theta$$
  $y = rsin\theta$   $z = z$  (1)

[그림 1]에 따르면 다음과 같은 식을 얻을 수 있다.

$$r = psin \oint \Theta = \Theta \quad z = pcos \oint$$
 (2)

(2)의 식을 (1)의 식에 대입하면 다음과 같은 결과를얻게 된다[12].

$$x = psin \oint cos\theta$$
  $y = psin\theta sin\theta$   $z = pcos \oint$  (3)

월드 좌표계를 통해서 동작이 생성된다고 가정할 때, 의도하는 동작에 따라 옮겨갈 공간상의 위치를 선분 단 위로 좌표상에 상세히 표시해야 하기 때문에 직관적이 지 않고 자연스러운 동작을 구성하기 위해서 많은 변수 를 고려해야 한다. 그리고 각 골격의 모양이 고정되어 있음에도 불구하고, 골격의 구조 대신 신체의 위치 좌 표를 중심으로 그려지게 됨으로써 골격을 구성하는 길 이, 폭, 굵기 등의 상대적 치수들이 어긋나버리게 된다. 모델 좌표계를 통해서 동작이 생성 될 때, 골격의 크기 인 p는 객체를 생성하기 위해서 정의되는 골격의 크기 로서 객체를 생성하는 순간 정의되고 Θ, ∮는 관절의 각도로서 동작을 생성함과 동시에 정의된다. 따라서 (3) 의 식을 이용하여 월드 좌표계에서 사용되는 직교 좌표 계를 두 각도를 사용하는 구면 좌표계로 대체할 수 있 고, 이는 에이전트의 신체 부분들의 회전과 이동을 이 용한 행렬을 통해 구현된다.

# 3.2 에이전트의 신체 구조 표현

인간의 주요한 신체 동작은 신체 부분들을 연결하는 관절의 움직임으로 나타낼 수 있다[13]. 따라서 본 연구 에서는 다양한 신체 동작들을 수행할 수 있는 에이전트 의 신체적 구조는 관절들을 중심으로 설계한다. 이 때, 역학적 구성을 체계적으로 표현하고 동작의 다양성을 수준별로 구현하기 위해 각 신체 부분들은 종속성에 따 라 계층적으로 구성한다. 에이전트를 구성하는 신체 부 분간의 연결 지점들은 각각 상응하는 관절과 일치하고, 각 관절에 대한 기본 동작은 관절에 연결된 신체 상위 부분과 하위 부분에 대해서 영향력을 가진다. 이 경우, 몸통과 직접적으로 연결 된 신체 부분 A는 A와 연결된 신체 부분 B의 움직임에 대해 직접적으로 영향을 준다. 즉, A와 몸통을 연결하고 있는 관절이 움직인다면, 자 동적으로 B도 A에 연동하여 움직이게 된다. 이 때, A는 B의 상위 부분이 된다. 각 관절의 회전 방향은 일반적 으로 두 개의 평면적 방향과 한 개의 비틀린 각도를 가 지도록 설계한다. 그리고 인간의 관절은 정해진 각도까 지만 회전할 수 있기 때문에 관절에 따라 최대 회전 각 도의 범위를 제한한다. [그림 2]는 에이전트의 기본적인 신체구성에 대해 대략적인 구조를 보여준다.

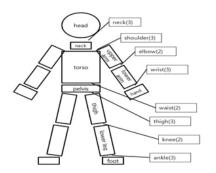


그림 2. 에이전트의 모델링

가상세계 내에서 사람이 아닌 '개'나 '고양이'같은 척 추동물들도 유사한 구조로 구현할 수 있다. 이 때 동물 은 사람과는 달리 관절의 회전의 크기와 방향이 다름을 고려해야 하고[13], 일반적인 사물의 동작에까지 확대 하여 적용하면 여닫이 문의 회전 기능이나, 미닫이 문 의 수평이동 기능 등을 통일적으로 구현할 수 있다.

### 3.3 에이전트의 역학적 요소의 계층적 구조

에이전트의 역학적 신체 구조는 [그림 2]가 보여주는 것과 같이 신체 부분들을 연결하는 관절들을 중심으로 한 계층 구조로 표현될 수 있다. 이 구조의 형태는 신체 의 연결이 사이클을 이루는 경우가 없는 사실을 반영하 여 순수한 트리 구조가 된다. 이 구조에 표현된 종속 관 계에 따라 상위 부분의 움직임은 관련된 하위 부분들에 계층적으로 전달된다. 이 때, 모든 신체 부분에 영향을 미치는 최상위 부분은 가장 복합적 행동이나 자세의 중 심이 된다. 실제로 걷기를 포함한 수많은 복합적 행동 들은 앉거나 허리를 숙이는 다양한 자세들의 중심이 되 는 골반 부분을 신체의 최상위 부분으로 선택한다.

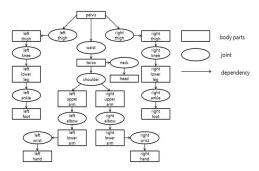


그림 3. 에이전트 신체에 대한 계층 구조

3차원 공간으로 표현되는 모델 좌표계 내에서의 동작은 신체 부분의 회전과 이동으로 표현된다. 3차원 공간을 표현하기 위한 오일러 각(Euler Angle)은 물체의 방위를 표시하기 위한 3개 각도의 조합이다. 오일러 각을 이용할 때, 회전 순서에 따라 물체의 회전된 최종 방위가 달라지며 일반적으로 ZYX(roll-pitch-yaw) 회전과 XYZ 회전이 주로 사용된다. ZYX 회전을 기준으로 한좌표계에 대한 물체 좌표계의 회전 R의 결과는 순차적으로 X축(yaw)을 중심으로 만큼, Y축(pitch)을 중심으로 만큼, 그리고 Z축(roll)을 중심으로 만큼 회전한다. 그리고 이동 변환 T는 좌표 값에 특정 값을 더한 값이다[14]. 다음은 관절의 회전 및 이동 즉, 관절의 동작에대한 행렬 연산이다.

$$\begin{split} r_x(\alpha) &= \begin{bmatrix} 1 & 0 & 0 \\ 0 \cos \alpha - \sin \alpha \\ 0 \sin \alpha & \cos \alpha \end{bmatrix} \\ r_y(\beta) &= \begin{bmatrix} \cos \beta & 0 \sin \beta \\ 0 & 1 & 0 \\ -\sin \beta & 0 \cos \beta \end{bmatrix} \\ r_z(\gamma) &= \begin{bmatrix} \cos \gamma - \sin \gamma & 0 \\ \sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \\ r_{xyz} &= r_z(\gamma) r_y(\beta) r_x(\alpha) \\ &= \begin{bmatrix} \cos \beta \cos \gamma & \sin \alpha \sin \beta \cos \gamma - \cos \sin \gamma & \cos \alpha \sin \beta \cos \gamma + \sin \alpha \sin \gamma \\ \cos \beta \sin \gamma & \sin \alpha \sin \beta \sin \gamma + \cos \alpha \cos \gamma & \cos \alpha \sin \beta \sin \gamma - \sin \alpha \cos \beta \\ -\sin \alpha & \sin \alpha \cos \beta & \cos \alpha \sin \beta \sin \gamma + \sin \alpha \sin \gamma \end{bmatrix} \\ m_i &= r_z(\gamma) r_y(\beta) r_x(\alpha) t_{xyz} \\ &= \begin{bmatrix} \cos \beta \cos \gamma & \sin \alpha \sin \beta \cos \gamma - \cos \alpha \sin \gamma & \cos \alpha \sin \beta \sin \gamma - \sin \alpha \sin \beta \sin \gamma - \cos \alpha \sin \beta \sin \gamma - \sin \alpha \sin \beta \cos \gamma \end{bmatrix} \end{split}$$

임의의 행렬인  $m_i$ 는 에이전트의 해당 신체 부위에 대한 이동 연산과 각 관절에 대한 회전 연산의 합이다. 관절(joint)을 구성하는 회전 행렬과 신체 부분(body part)을 구성하는 이동 행렬의 곱은 하나의 신체 부분을 움직이는 행렬을 구성하는 요소들이며, 신체의 계층적 구조에 따라 하위 신체 부분이 자신을 포함하고 있는 모든 상위 신체 부분의 움직임에 영향을 받는다.  $m_k$ 를 임의의 신체 부위의 위치 행렬이고 을 최상위 신체 부위의 위치 행렬이라고 할 때, 해당 신체 부위의 위치

는 역학적으로 연결된 모든 상위 신체 부위들의 위치 행렬들의 곱이 된다. 즉,  $m_k = m_{i-1} \times m_{i-2} \times m_{i-3}$ ... $m_1$ 이 된다.

이러한 모델 좌표계를 통한 구현 방식은 직관적이고 편리하지만, 충돌을 위한 신체 부분의 위치를 획득함에 어려움이 있다. 따라서 신체 부분의 실체 위치는 모델 좌표계 내에서 구현되는 에이전트가 효율적으로 외부 객체와 상호 작용을 하기 위해 월드 좌표계로 변환되어 야 한다.

### Ⅳ. 기본 동작

# 4.1 동작의 분류 및 구조

에이전트의 움직임은 기본 동작, 복합 동작 그리고 행동으로 분류한다. 기본 동작은 한 개의 관절의 움직 임을 지칭하고, 복합 동작은 두 개 이상의 관절의 움직 임으로 이루어지는 움직임을 가리킨다. 그리고 행동은 특정 효과를 달성하기 위해 유기적으로 조합된 기본 동 작들이나 복합 동작들의 순차적 배열을 의미한다. 기본 동작은 모든 복합 동작이나 행동을 수행하기 위한 기본 요소가 된다. 결과적으로 모든 에이전트의 행동들은 기 본 동작들을 반복적으로 사용함으로써 이루어지게 된 다. 역으로 기본 동작들을 조합함으로써 새로운 동작들 을 조립식으로 생성해낼 수 있게 된다. 이렇게 생성된 복합 동작 또는 행동을 한 개의 기능으로 새롭게 정의 할 수 있다. 나아가 에이전트가 정한 목적을 달성하기 위해서는 여러 가지 동작들을 사용하여 주어진 상황을 목표상황으로 변화시키게 된다. 이 때 동작들을 시간적 으로 배열하는 과정은 기본 동작을 조합하여 복합 동작 을 만드는 과정의 연장이라고도 할 수 있다.

#### 4.2 동작의 모델링

(4)

각 동작의 결과인 목표 상태는 관련된 관절을 움직이는 함수에 의해 결정된다. 이 함수는 관절의 각 회전축의 개수에 상응하는 인자들을 가지고, 각 인자의 변역은 관절의 상태가 변할 수 있는 범위를 규정한다. 이러한 관절들의 동작이 시각적으로 자연스럽게 구현됨과

동시에 각 관절에 대한 테이터를 참조한다. 예상치 못한 상황에 대응하기 위해서 이러한 과정은 실시간으로 진행되고 이 함수는 새로운 함수 인자가 입력되면 그에 대응하여 목표 상태를 변경한다. 이러한 함수들은 관절의 초기 각도 상태를 참조하고 목표하는 각도에 대한 입력 인자로 가짐으로써 시각적으로 보이는 신체 부분이 이동할 동선을 생성한다. 그리고 동작이 종료되면 목표 상태가 다음 동작의 초기 상태가 된다. 함수  $f(a_i)$ 가 각 관절의 목표 상태를 결정하고 시간(t)이 진행되면, 에이전트의 내부 테이터인 관절의 각도  $m_i$ 는 점진적으로 변화된다. 즉 모든 신체 부분을 제어하는 함수들은 다음과 같이 구성된다.

$$\prod_{i=1}^{n} (f \colon a_{i} \to m_{i})$$

각 함수들은 관절의 각도를 시간에 따라 점진적으로 변화시킴으로써 독립성을 가진다. 따라서 관절의 동선 을 실시간으로 변경함으로써 이를 포함하는 동작 및 행 동이 다양한 상황에 대응할 수 있는 기반을 마련한다. 또한, 동작 함수는 사용자가 동작을 조합하기 위한 도 구의 최소 단위 및 병렬 처리의 기본이 된다.

#### 4.3 동작의 구현

6

관절의 시작 상태로부터 목표 상태에 도달하는 과정은 애니메이션에서와 같이 정적인 피사체나 사물 등을 시각의 잔상을 이용하여 동적으로 표현해준다. 캐릭터 애니메이션을 제작함에 있어서 키 프레임 방식은 캐릭 터의 골격을 편집 프로그램을 이용하여 조작함으로써 키 프레임을 생성하고 보간함으로써 동작을 시각적으로 생성한다. 그리고 모션 캡처 방식은 배우의 움직임에 따라 관절과 신체 부분의 위치에 대한 데이터를 생성하고 이를 골격과 결합하여 움직임을 구현한다[15].

에이전트의 행동을 시각적으로 구현하기 위한 이러한 기법들은 월드 좌표계 또는 모델 좌표계를 기반으로 구현될 수 있다. 신체의 중요한 관절의 위치에 대한 값을 이용하여 신체 부분의 길이와 관절의 각도를 연산하는 모션 캡처 방식과 월드 좌표계를 통한 구현 방법은 흡사하다. 그러나 하드웨어의 도움을 받을 수 없는 가

상 세계 내에서는 신체 부분의 길이에 대한 값을 지속 적으로 입력을 받을 수 없다. 따라서 자연스러운 신체 의 골격의 길이를 유지하기 위해서 애니메이션이 진행 되는 동안 동작을 그려주는 연산을 함에 있어서 월드 좌표계를 기반으로 구현하는 것은 비효율적이고 구현 된 동작의 표현이 직관적이지 못하다. 반면에 주어진 골격의 크기와 관절의 각도를 이용하여 신체 부분의 위 치를 결정하는 모델 좌표계를 통한 에이전트의 동작 구 현 방식은 관절을 조절하고 신체 부분을 움직임으로써 쉽게 동작을 생성하게 해준다. 이는 키 프레임을 설정 하고 동작을 보간 및 생성하는 키 프레임 기법과 흡사 하다. 따라서 본 연구에서 목표로 삼는 다양한 동작의 효율적 생성 및 표현을 위해 에이전트는 동작을 실행함 과 동시에 목표 상태를 키 프레임으로 설정함으로써 관 절의 각도를 변경하여 동적인 움직임을 표현한다. 일반 적으로 에이전트가 목표 상태에 도달하는 과정을 묘사 하기 위한 프레임의 수는 시작 상태와 목표 상태의 간 격에 비례하여 증가한다. 같은 간격에 대해 프레임의 수가 증가한다면 행동은 더 정밀하게 표현됨을 의미한다.

# V. 행동

#### 5.1 행동의 구조

소수의 기본 동작들은 미리 정의되고, 그것들을 병렬적으로 조합하여 복합 동작들을 생성하고, 그들을 순차적으로 사용함으로써 기본 행동들을 구축한다. 행동은하나의 기본 동작 또는 복합 동작을 반복하거나, 또는복수의 기본 동작 또는 복합 동작을 순차적으로 실행한다. 기본 동작은 n개의 상(phase)으로 구성된다. 어떤행동을 구성하는 각 동작 A,B,C…N은 다음과 같은 열로 표시된다.

그러면 해당 행동은 아래와 같이 같은 복합적 상인

 $\pi_i$ 로 나타낼 수 있다.

$$\begin{split} \pi_1 &= a_1, b_1, c_1 \cdots n_1 \\ \pi_2 &= a_2, b_2, c_2 \cdots n_2 \\ \pi_3 &= a_3, b_3, c_3 \cdots n_3 \\ &\vdots \\ \pi_n &= a_n, b_n, c_n \cdots n_n \end{split}$$

행동은 다음과 같은 식에 따라 계획된다.

$$S(t+\Delta t) = S(t) + \prod_{i}^{M} \left(\sum_{j}^{N} m_{j}^{i}, such \, as \, A_{s} \supset m_{j}^{i}\right) \tag{5}$$

모든 행동을 구성하는 최소 단위는 상의 구성 요소인 기본 동작이다. 기본 동작을  $m_i^i$ 이라고 할 때, 임의의 행동인  $A_s$ 에 포함된 모든 기본 동작의 합은  $\pi_n$ 이 된 다. 는 시간이 진행됨에 의해 순차적으로 실행되는 의 집합이다. 즉, 위 식은 임의의 행동에 포함된 모든 순차 적 단계에 존재하는 기본 행동의 집합을 의미한다. 에 이전트는 특정한 목적에 따라 기본 동작 또는 복합 동 작을 포함한 상을 순차적으로 실행하거나 반복 및 지속 함으로써 행위를 수행한다.

## 5.2 행동을 위한 동기화 및 계획

복합 동작은 여러 관절들의 동작이 유기적으로 조합 되어 행해진다. 이러한 복합 동작들을 자연스럽게 구현 하기 위해서 복합 동작을 구성하는 각 신체 부분들의 움직임이 동기화되어야 한다. 이를 위하여 복합 동작을 구성하는 관절들의 시작 상태와 목적 상태의 차이에 비 례하여 해당 관절이 움직이는 각도를 조절한다. 즉, 한 에이전트의 모든 행위들의 움직임은 동일한 시간 내에 실행된다.

대표적 복합 동작인 '걷기()'를 실행하기 위한 에이전 트의 최상위 부위의 움직임과 그에 관련된 기본적인 물 리 현상들 또는 다른 객체와 상호 작용을 위한 함수를 살펴본다. 구체적으로, [그림 4]은 이동에 관한 함수를 이용함으로써 '걷기()'가 어떠한 방식으로 동작하는지 보여준다. 다리 R가 지면을 내딛고 다리 L이 이동하고

골반 부분이 함께 이동한다. 따라서 계층적 구조에 의 해서 모든 신체 부분은 길이 3의 거리만큼 이동하게 된 다. 길이 1은 다리 R가 관절의 회전에 의해 이동하면서 골반의 위치를 기준으로 발의 위치가 지형과 직각에 위 치할 동안의 거리이고, 길이 2는 다리 L이 지면을 내딛 고 다리 R가 골반의 위치와 직각인 지면으로부터 이동 한 거리이다. 다리 R가 관절의 회전에 의해 이동한 거 리가 길이 1과 길이 2의 합보다 짧거나 길다면 지면의 내딛는 발의 속도가 몸의 이동보다 빠르거나 느리게 되 고 행동이 부자연스럽게 된다. 이와 같이 모든 행동은 'speed'라는 인자를 참조하여 행동의 속도와 이동 속도 를 동기화한다.

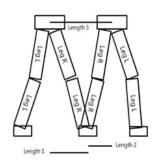


그림 4. 이동에 의한 움직임

'걷기()'와 같은 하나의 반복되는 복합적 행동은 일정 한 패턴을 가진다. '걷기()'는 단계별로 신체 부분의 동 작을 실행시킴으로써 여러 집합을 가진다. [표 1]은 '걷 기()' 행동이 계획에 포함된 각 단계별로 동작하는 방법 을 보여준다.

표 1. 계획에 의한 '걷기()'의 구조

신체 부분	phase 1	phase 2	phase 3	phase 4	
left arm	<b>→</b>	<b>←</b>	<b>→</b>	←	
right arm	-	<b>→</b>	<b>←</b>	<b>→</b>	
left leg	<b>←</b>	<b>→</b>	<b>←</b>	<b>→</b>	
right leg	<b>→</b>	-	<b>→</b>	-	

8

이러한 단계별로 존재하는 집합은 하나의 복합적 상이라고 볼 수 있다. 개별적으로 존재하는 각 동작들은 초기 상태에서 목표 상태에 도달하는 시간이 각각 다르다. 그러나 한 단계 내에서 행동을 구성하고 있는 동작들은 동일한 시간에 목표 상태에 도달해야 한다. 즉, 'speed' 인자에 따라 신체 부분의 움직임을 조절한다.

동작의 움직임은 초기 관절 각도를 기준으로 목표 상태에 대한 방향성을 가진다. 한 신체 부분에 대해서 동일한 시간에 다양한 방향성을 가진다면, 정밀함에 중점을 둔 행동이라고 볼 수 있다. 예를 들어, 일반화된 '걷기()' 동작이 표현되는 과정에서 팔과 다리를 제어하는 관절의 각도는 초기 상태에서 목표 상태에 도달하기까지 한 개의 회전 축의 이동으로 표현될 수 있다. 그러나실제 사람은 동작 과정에서 시간이 흐름에 따라 크게회전하는 회전축과 미세하게 회전하는 다른 회전축들이 함께 동작한다. 즉 계획에 포함되는 다양한 방향성을 가지는 동작 집합의 개수에 따라 행동의 정밀도가결정된다. 행동의 정밀도인 x는 동작 집합과 다음과 같은 관계를 가진다.

$$x = \frac{\sum_{i=1}^{n} d_i \left(\sum_{j=1}^{n} m_j^n\right)}{T} \tag{6}$$

 $d_i$ 는 행동이 포함하는 동작의 방향성의 최대 변화 개수이고 행동의 단계의 수를 의미한다. 각 단계를 구성하는 세트인 동작들은 각 다른 방향성의 변화 개수를 가질 수 있고  $m^i_j$ 는 각 동작들이 가지는 방향성의 변화 개수이다. 행동에 사용되는 신체 부분마다 요구하는 정밀도가 다를 수 있기 때문에 방향성이 변하지 않는 신체 동작은 목표 각도를  $d_i$ 의 개수만큼 분할하여 동작한다. 즉, 행동이 포함하는 동작의 방향성의 최대 변화 개수는 행동을 표현하는 최소 단계의 수를 결정하는 기준이 된다.

다른 방향성을 가지는 각 신체 부분의 동작이 다른 시간에 종료되는 반복되는 행동을 구성한다고 가정할 때, 동작을 표현하는 단계가 더 분할되어야 한다. 예를 들어 먹으면서 걷는 행동이 있다고 가정하면, 음식을 먹는 기능을 하는 팔은 걷는 행동을 나타내는 [표 1]의 신체 구조의 동작과는 달리 한 동작이 기능적으로 종료되는 시점이 다르다. 동일한 단계에 속한 동작들은 동일한 시점에 종료되기 때문에 행동에 포함된 모든 동작이 소요되는 시간 t는 각 상에서 지연되는 시간인  $\Delta t$ 의 배수가 된다.

#### 5.3 행동의 재활용

애니메이션을 효율적으로 제작하기 위해서 동작을 재사용하는 기법들은 3D 저작 도구의 그래프(graph) 또는 에디터(editor)를 이용하여 동작을 편집하거나 모션 에디터(motion editor) 등을 통해서 동작을 재활용한다[16]. 그러나 이러한 기법들은 대부분 모션 캡처 방식을 기반으로 사용된다. 모션 캡처는 아주 낮은 레벨에서부터 데이터를 생성하기 때문에 데이터를 재활용하기가 어렵고 효율적이지 못하다. 또한 가상세계 내에서발생되는 예기치 못한 상황에 대처하여 동작들을 효과적으로 구현할 수 있어야 한다. 본 연구에서 행동은 각신체 부분들의 동작으로 구성되고 신체 부분들의 움직임은 행동의 정밀함에 따라 다음과 같이 분해된다.

$$A_i = \langle P_1, P_2, P_3, \cdots P_n \rangle$$

모션 캡처와 같은 기존의 방식은 동작의 재활용이 불가능하기 때문에, 신체 부분의 움직임이 조금 수정되더라도 모든 신체 부분의 움직임을 생성해야 한다. 따라서 이러한 기법은 다양한 행동을 구현함에 있어서 매우비효율적이다.

동작을 재활용하는 기법은 각 신체 부분의 동작을 개별적으로 생성하고 조합함으로써 기존의 행동을 재생성하지 않고, 행동을 생성하는 과정을 효율적으로 만든다. 예를 들어 신체 부분  $|P_1|,|P_2|,|P_3|$  이 각각 3개의 동작을 가지고 있다고 가정하면, 총 9개의 동작을 구현함으로써 27개의 행동을 구현할 수 있다. 그러나 동작을 재활용하는 기법은 이러한 개별적인 동작을 조합하는 과정이 필요하고, 이를 최소화시킴으로써 효율성을 높일 필요가 있다.

### 5.4 병렬 및 병행 행동에 대한 조합

복합 동작은 각 관절을 제어하는 함수들을 통해 각동작의 목표 상태를 설정하고, 공통되지 않는 동작 (disjoint motions)들을 실시간으로 병렬 실행함으로써 실행된다. 병렬 행동을 실행함에 있어서 이와 같이 공통되는 신체 부분의 동작들은 동시에 요구된다. 같은 시간에 같은 신체부분이 여러 동작들을 동시에 수행할수 없기 때문에 우선순위를 정할 수밖에 없다. [표 2]에서 보는 바와 같이 신체 부분의 동작을 필수적 (essential) 동작과 선택적(optional) 동작, 그리고 불필요한(null) 동작으로 분류한다.

필수적 동작은 실행하는 행동에 있어서 반드시 포함 되어야 하는 동작을 의미하고 선택적 동작은 보조적 역 할로서 필수적이지 않은 동작을 의미한다. 불필요한 동 작은 미세하게 움직이거나 또는 움직임이 없는 상태이다.

표 2. 중요도에 의한 동작의 분류

행동2 행동1	Null	Optional	Essential
Null	불용	행동2	행동2
Optional	행동1	선택	행동2
Essential	행동1	행동1	병렬행동불가

예를 들어, '걸으면서 먹는다'라는 병렬 행동이 있다고 가정한다면 조합되는 두 행동은 'walk'와 'eat'이다. 'walk'라는 행동에 있어서 필수적 동작은 이동에 관련된 다리 부분의 움직임이고 팔 부분의 움직임은 보조적역할로서 선택적 동작이다. 그리고 움직임이 거의 없는 몸통 부분과 머리 부분은 불필요한 동작으로 정의한다. 'eat'이라는 행동에 있어서 오른쪽 팔 부분과 입 부분의움직임이 필수적 동작으로 정의하고 다른 부분은 불필요한 동작이라고 정의한다. 이러한 병렬 행동은 다리부분과 왼쪽 팔 부분은 'walk'에 포함된 동작을 실행하고 오른쪽 팔 부분과 입 부분은 'eat'에 포함된 동작을실행함으로써 생성된다.

실제로 에이전트의 하나의 신체 부분에 대한 복수의 동작들은 동시에 발생할 수 없다. 따라서 각 신체 부분 을 움직이는 기본 동작으로 구성된 복합 동작들은 중요 도에 따라 동작을 제어하는 각 변수에 제어됨으로써 조 합된다. 기본 동작들은 독립적으로 동작하므로 신체의 계층적 구조에 상관없이 조합이 가능하다. 그러나 병렬행동을 조합하기 위해서는 해당 행동을 구성하는 신체부분에 대한 동작들의 중요도를 결정해야 하고, 중요도를 비교하기 위한 복합 동작들은 동일한 단위의 움직임으로 구성되어야 한다. 이를 위해서 행동을 구성하는 각 복합 동작들은 하나의 기능을 표현한다. 기능별로구별한 복합 동작들은 팔과 다리, 몸통, 머리로 나뉠수도 있고, 크게 상체와 하체로 나뉠 수도 있다. 즉 복합동작은 조합하는 범위는 관계가 없으나, 이러한 동작들이 기능적 의미를 가질 수 있는 단위로 구성되어야 한다. 즉, 복합 동작을 조합하는 단위는 다음과 같은 조건을 포함한다.

$$M = (P_n, P_{n-1}, P_{n-2}, P_{n-3} \cdots P_{n-k})$$

 $P_n$ 는 팔, 다리와 같은 기능적 의미를 가지는 계층 구조에 포함된 신체 부분이다. 계층 구조에 포함된 신체들은 힘이 역학적으로 전달하는 연결 고리가 존재하고, 복합 동작을 조합하는 단위는 이러한 연결고리를 통해 연결된 신체 부분들을 포함해야 한다. 즉 계층적구조에 존재하는 신체 부분의 위치에 따라 n이 결정될 때, 기능적인 역할을 하는 신체 부분들은 임의의 구간 인  $P_n$ 부터  $P_{n-k}$ 까지 연결되어 있다.

이러한 단위를 구성하는 신체 부분의 개수는 신체의 구조를 구성하는 방법에 따라 달라진다. 예를 들어, 몸 통을 하나의 신체 부분으로 구성할 수도 있고, 또는 허 리의 다양한 움직임을 위해서 허리를 여러 부분으로 나 누어 다수의 세분화된 계층적 구조로 표현할 수도 있 다. 따라서 움직임의 정밀함에 따라 계층적 신체 부분 들을 구성한다. 단, 신체를 구성하고 동작을 수행하는 부분이 증가함에 따라 복합 동작 및 행동의 생성 및 조 합이 더 어려워지게 된다.

병행 행동은 두 행동이 동작할 신체 부분의 중요도가 같으나 제한된 시간 내에 작업을 끝내야 한다면, 이러 한 두 행동을 번갈아가며 실행한다. 병행 행동을 구현 하는 방식은 반복되는 복합 동작과 다르지 않으나 두 동작이 한 행동으로 조합됨으로써 표현되는 것이 아닌 행동과 행동의 조합으로 표현된다. 즉 동작을 직접 조합하고 계획하여 행동을 생성하지 않고 중요도에 따라 병렬 행동이 불가능한 신체 부분에 대한 두 동작들만을 병행한다.

상황 변화에 대응하기 위한 행동의 조합 및 교체는 이러한 정보를 기반으로 하는 병렬 구조에서 이루어진다. 해당 에이전트의 실행될 예정인 행동이 다수라고가정하면, 각 공통된 신체 부분의 동작에 따라 병렬 행동으로 실행되거나 행동의 교체가 결정된다. 따라서 병렬 행동의 실행으로 인해 어떠한 동작이 추가되거나 또는 갑작스러운 행동의 전환이 발생되더라도, 자연스럽게 다음 행동을 즉시 실행할 수 있다. 병렬 행동을 조합하거나 행동을 전환하는 과정은 다음 식에 의해 이루어진다.

$$S_k = \sum_{i=1}^{n} p_i \left( \prod_{j=1}^{n} \left( \sum_{j=1}^{n} f(a_j^q) \right) \right) + S_{k-1}$$
 (7)

가상 세계 내에서 존재하는 상황들은 시간이 흐름에 따라 모든 에이전트들이 행동을 실행할 때 가지는 복합적 상의 집합에 의해 변화한다. 에이전트의 집합  $p_i$ 는 계획에 따라 순차적으로 구성된 단계인 로 구성되어 있다. 이러한 단계는 시간에 따라 목표 상태인  $a_j$ 를 달성하기 위한 점진적으로 실행되는 동작 함수들로 이루어진다. 각 단계는 모든 동작 함수들의 목표 상태가 충족되면 순차적으로 다음 단계의 함수인  $f(a_j)$ 가 동작한다. 즉, 모든 에이전트의 행동을 실행하기 위한 계획은독립적인 동작의 생성을 바탕으로 이루어진다.

# VI. 동작에 따른 상호 작용

가상 세계에는 많은 수의 객체들이 존재하고 그들은 일상적으로 다른 객체와 상호 작용하면서 업무를 수행한다. 이러한 상호 작용은 객체들이 공간적으로 서로 근접해지거나 나아가 접촉내지 충돌하는 것을 전제로 발생한다. 일반적으로 실시간 진행이 필수불가결한 요소인 게임에서도 계산량이 상대적으로 많은 물리 엔진

을 사용한다. 이러한 물리 엔진은 실제 힘과 질량의 관계식으로부터 물체의 이동 방향과 위치를 계산하여 자연스러운 움직임을 표현한다[17]. 그러나 정교한 물리엔진은 상황을 표현함에 있어서 연산이 많으므로 비효율적이다. 따라서 물체의 충돌을 표현할 때, 충돌을 감지하기 위한 기본적인 연산을 제외하고는 기존의 물리적 충돌 검사보다는 애니메이션 기법을 통한 물체의 움직임을 묘사하는 것이 효율적이다.

가상 세계 내에서 공간을 점유하는 객체들은 능동적 객체와 수동적 객체로 분류할 수 있다. '사람'과 같이 능 동적인 객체는 스스로 하는 행동뿐만 아니라, 다른 능 동적, 수동적 객체들의 대응을 유도한다. '책', '창문' 과 같이 수동적인 객체는 능동적인 객체의 메시지를 전달 에 의해 받아 '펴다', '창문을 열다' 와 같은 기능을 수행 한다. 이러한 상호 작용은 모델 좌표계로부터 변환된 월드 좌표계의 객체간의 상호 작용으로부터 발생하고, 능동적 객체의 목적에 따라 주체적 객체가 메시지를 다 른 객체에게 전달함으로써 동시에 알맞은 행위를 실행 한다. 이를 위해 가상 세계 내에서 존재하는 모든 객체 들의 행동을 동시에 처리할 수 있는 멀티태스킹 (multi-tasking)과 같은 병렬 처리가 필요하다. 단, 실행 될 행동들에 대한 필요한 시간은 공평하게 분배되므로 실제 멀티태스킹과 같은 작업 전환(task switching) 전 략은 필요하지 않다. 이러한 구조는 집합을 구성 요소 로 하는 다 수의 리스트(list)로 구성되고, 각 세트는 객 체들이 상호 작용 및 행동을 하기 위한 정보인 객체를 식별하기 위한 상수, 상호 작용하기 위한 대상 객체의 정보, 실행되는 행동의 목표 상태 등을 포함한다.

## Ⅶ. 구현 및 실험 결과

본 시스템의 목표는 기본 동작을 이용하여 다양한 효율적인 행동들을 효율적으로 생성하는 데 있다. 따라서 동작을 재활용하는 방법을 통해 행동을 쉽게 생성하고, 동적인 상황의 변화에 적응하기 위한 행동 전환과 행동조합을 실시간 실행 및 동작 변화를 통해 구현하였다. 구현을 위한 도구는 Visual studio 2008과 OpenGL implementation을 사용하였고, Microsoft Access 2007

을 ODBC를 통해 데이터를 구축하였다.

인간의 동작을 구현하는 시스템에서 먼저 고려해야 할 점은 인간의 동작이 온톨로지와 연계됨을 바탕으로 변화된다는 점이다. 따라서 갑작스러운 환경 변화에 의 한 객체 반응 및 변화를 고려하고, 기본적인 온톨로지 를 구현함으로써 계층적 구조에 따른 객체 구현을 기반 으로 설계하였다. 따라서 최종적인 가상 세계를 구현하 기 위한 기본 바탕이 되는 도구가 될 것이다. 다만, 현 재 시스템은 온톨로지에 의한 인간의 의도나 욕구, 반 사적 반응을 고려하지 못하기 때문에 수동적인 입력 방 식이 필요하다. 입력 도구는 능동적, 수동적 객체 선택, 행동 조합 선택, 행동에 필요한 인자들을 입력한다. 다 음은 기본 동작을 조합하고 반복함으로써 동작하는 '걷 기'다.

'걷기'와 같은 에이전트의 이동이 요구되는 행동은 입 력 도구로부터 이동할 목표 위치를 요구한다. 에이전트 가 목표 위치로 이동하면 각 신체 부분의 관절들이 계 획에 따라 목표 상태에 도달하기 위해서 동작한다. 이 때, 팔과 다리의 초기 상태와 목표 상태의 간격이 서로 다름에도 동기화됨으로써 함께 동작한다.

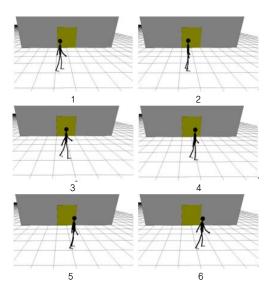


그림 5. 재활용에 의한 '걷기' 행동

다른 객체와 상호 작용이 필요한 행동의 경우, 행위 를 구현하는 해당 객체는 다른 객체의 식별 번호를 필 요로 한다. 식별 번호를 기반으로 객체를 구별하고, 메 시지를 전달함으로써 상대 객체의 행위를 유도한다. 능 동적 객체는 이러한 행위를 통해 다른 능동적, 수동적 객체들과 상호 작용을 행한다. [그림 6]은 능동적 객체 인 에이전트가 수동적 객체인 '집'에 포함된 '문을 열다' 라는 기능을 전달된 메시지에 의해 실행하는 장면이다.

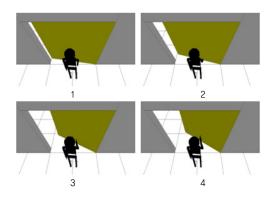


그림 6. 에이전트와 객체의 상호 작용

가상 세계 내에서 실현되는 갑작스러운 상호 작용 및 사건 발생과 관련되는 에이전트의 행동 및 동작 교체는 가상 세계 구현에 있어서 중요한 작업이다. 병렬 및 병 행 행동은 이러한 작업을 위한 효율적 행위 방법을 보 여준다.

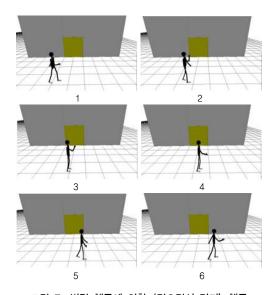


그림 7. 병렬 행동에 의한 '걸으면서 먹기' 행동

병렬 행동은 신체를 기능적으로 분류하여 조합하는 행동이다. 이러한 행동은 에이전트가 일정한 시간 내에 여러 작업을 진행할 수 있게 만든다. 제한된 시간 내에다 수의 동작들을 수행해야 할 때 병렬 행동은 병행 행동으로 대체된다.

# Ⅷ. 결론

본 논문은 가상 세계 내에서 에이전트들이 행하는 다 양한 행동을 효율적으로 생성하기 위해 구조들과 구현 방법을 개발했다. 이러한 방법은 소수의 기본 동작들을 정의하고 조합함으로써 복합 동작들로 구성된다. 에이 전트의 신체 구조는 관절들을 중심으로 구성되고, 기본 동작의 움직임을 위한 함수를 통해 애니메이션을 생성 함으로써 기본 동작 및 복합 동작들은 실시간으로 실행 될 수 있다. 즉, 기본 동작들은 병행 행동 및 병렬 행동 을 구성하기 위해 재사용되고, 이를 이용하여 에이전트 들이 주위의 환경에 즉시 반응할 수 있도록 만든다. '건 기'와 같은 복합 행동은 복합 동작을 반복함으로써 실 행되는 것을 보여주고, '문을 열다'와 같은 상호 작용에 대한 복합 행동은 에이전트가 메시지 전달을 통해 '문' 과 같은 외부 객체와 상호 작용할 수 있음을 보여준다. 이러한 기본 동작들을 다양하게 조합함으로써 에이전 트는 복잡한 동작들을 추가하지 않고도 여러 가지 목표 들을 구현할 수 있다. 이러한 접근법은 '걸으면서 먹기' 와 같은 병렬 행동을 구성할 때 독립적 개별 행동들을 순위에 따라 조합하여 구현할 수 있음을 보여주었다. '걸으면서 먹기'는 '걷기'와 '먹기', 두 행동이 어떻게 병 렬 행동으로 구현되는지 보여준다. 본 연구는 사실감을 희생하는 대신 다수의 복잡한 행동들을 효율적으로 구 현했고, 몇 개의 행동을 구현함으로써 제시한 방법의 실현 가능성을 증명했다.

#### 참고문 헌

[1] W. B. Lee, "Technical Trends in Virtual Reality", 한국콘텐츠학회논문지, 제7권, 제4호,

- pp.29-35, 2009.
- [2] T. W. Han, "Research on the Design of Game Characters Utilizing Humanoid Robot", 정보디 자인학연구, 제19권, 2012.
- [3] 양정모, "적응형 NPC 생성을 위한 FSM의 동적 활용 방안", Journal of korea multimedia society, Vol.11, No.9, pp.1258-1266, 2008(9).
- [4] H. Kim and H. Yoon, "The control of character's behavior by using FSM-based probability estimation in games," Journal of korea multimedia society, Vol.8, No.9, pp.1269-1281, 2005(9).
- [5] J. E. Lird, "Using a computer game to develop advanced AI," IEEE, Vol.34, Issue.7, pp.70–75, 2001(7).
- [6] 이동엽, "심리학적 도구 '5요인 성격 특성'에 의한 소셜 게임 연구", 만화애니메이션연구, 제29호, pp.129-149, 2012.
- [7] 이인환, 상황 인식 온톨로지의 제사용성 향상을 위한 상위 온톨로지 추출 및 도메인 분류 기법 연 구, 부산대학교, 2013.
- [8] G. H. Jung, The implementation of Spatio-Temporal graph to represent situation in the virtual world, Graduate School Kyungpook National University, 2012.
- [9] B. H. Lee, etc. "Design and implementation of the MPEG4 player on the J2ME", 한국정보과학 회, Vol.29, No.2, 2002
- [10] T. J. Choi, etc. Abstract of digital motion capture system and 3D game character animation application, Kongju National University, 2006.
- [11] S. Richard, J. Wright, and B. Lipchak, *OpenGL* super bible, Information Publishing Group, 2005.
- [12] 이민영, Studies on the coordinate systems, 한 남대학교 교육대학원, 2008.
- [13] I. K. Lee and J. H. Park, "Representation of 'Walk' for quadruped animal based on primitive

- action and class inherit," Journal of the korea contents association, Vol.6, No.5, pp.85-95, 2006.
- [14] D. James, Representing Attitude: Euler Angles, Unit Quaternions, and Rotation Vectors, Stanford University, 2006.
- [15] S. H. Ryu, etc. "3D game character animation pipe-line to improve utilization of motion capture", 한국콘텐츠학회논문지, 제8권, 제7호, 2008.
- [16] S. H. Song and C. S. Kim, "A study on 3D motion data reapplication model", 디지털디자인 학연구, 제4권, 2002.
- [17] Y. J. Ha and K. J. Park, "A study on game physics engine focused on real time physics", 한국게임학회지, 제9권, 제5호, pp.43-52, 2009.

#### 저 자 소 개

### 최 준 성(Jun-Seong Choi)

정회원



• 2011년 : 대구대학교(공학사)

• 2014년 : 경북대학교(공학석사)

• 2014년 ~ 현재 : 경북대학교 전

자공학과 박사 과정

<관심분야>: 인공지능, 가상현실

#### 박 종 희(Jong-Hee Park)

정회원



• 1979년 : 서울대학교(공학사)

■ 1981년 : 한국과학원(공학석사)

■ 1990년 : Univ. of Florida(공학 박사)

• 2003년 ~ 현재 : 경북대학교 전 자공학과 교수

<관심분야> : 멀티미디어 응용, Computer Aided Education, CAD/CAM, 지능형 정보 시스템, 분산데 이터 처리 시스템