

다중 레이어를 이용한 모션캡처 수정에 관한 연구

김기홍[†], 최철영^{**}, 채일진^{***}

요 약

모션캡처 데이터의 떨림 현상 수정이나 변형을 위한 기술 개발이나 애니메이션 타이밍 시트에 맞게 모션 캡처 데이터를 수정하는 것은 어려운 문제가 산적해 있다. 기존의 모션캡처 데이터 수정 방식은 능숙한 애니메이터의 키프레임 애니메이션 작업과 같은 시간이 소요되거나 혹은 더 많은 수정 시간이 걸리는 문제점이 있었다. 이런 문제점은 키애니메이션 데이터 노드와 직접적인 블렌드 레이어와 대치 레이어 노드 생성을 통해 보다 효과적인 해결 방식을 만들 수 있다. 본 논문에서는 애니메이션 노드에 직접적인 연결을 위한 새로운 애니메이션 레이어 노드 생성으로 진행된 애니메이션 데이터 수정 없이 비선형 방식으로 애니메이션 데이터의 수정을 가능케 하는 방식으로 제시한다. 기술의 구현은 'Maya' API를 활용할 것이고 연구범위는 영화 VFX와 애니메이션 제작에 사용되는 'Maya' 3D 소프트웨어로 제한하였다. 본 연구 결과물은 기존의 비선형 방식보다 직관적이며, 애니메이션 클립을 만들 선행 작업이 필요 없고, 떨림수정과 키프레임 추출이 가능하며, 특히 타 프로그램과의 호환이 직접적인 레이어 노드를 생성하여 모션캡처 데이터의 수정을 가능하게 한다. 마지막으로 기존의 애니메이션 수정방식들을 살펴보고 비교, 분석해 본다.

The Revision of Motion Capture Data using Multiple Layers

Ki hong Kim[†], Chul Young Choi^{**}, Eel Jin Chae^{***}

ABSTRACT

There are still many difficulties in developing techniques for modifying or transforming the flickering of motion capture data or in modifying motion capture data in such a way that suits the animation timing sheet. There is a problem in the existing method of modifying motion capture data. It requires almost same time as in the key frame animation work by a very skilled animator or even more time in modifying. It is believed that this kind of problem can be a basis for a more effective problem-solving method through creating the key animation data node and direct blend layer and replacement layer nodes. This study presents a new method which enables to modify animation data in a nonlinear way without modifying the existing animation data by creating an animation layer node for a direct connection to the animation node. 'Maya' API will be utilized in order to realize this method and the research range will be limited to 'Maya' 3D software which is generally used in motion picture and animation films. According to the results of this study, the new method is much more intuitive than the nonlinear one and does not require the preceding working of making animation clips. In addition, it has enabled to modify flickering and to extract key frames, and due to the compatibility with other programs, it has been possible to modify motion capture data by creating a direct layer node. Finally, in this study, the existing method of modifying animation will be examined, compared and analyzed.

Key words: 3D animation(3D 애니메이션), Motion Capture(모션 캡처), Layer(레이어), Blend(블렌드)

※ 교신저자(Corresponding Author): 채일진, 주소: 부산시 사상구 주례2동(617-716), 전화: 051)320-1975, FAX: 051)320-1513, E-mail: cinetree@gdsu.dongseo.ac.kr
접수일: 2008년 8월 20일, 완료일: 2009년 4월 30일

[†] 정회원, 동서대학교 디지털콘텐츠학부 부교수

^{**} 정회원, 동서대학교 디지털콘텐츠학부 조교수

(E-mail: freechoi@gdsu.dongseo.ac.kr)

^{***} 정회원, 동서대학교 디지털콘텐츠학부 조교수

(E-mail: cinetree@gdsu.dongseo.ac.kr)

※ 이 논문은 중소기업청의 산학공동기술개발 지원 사업을 통해 개발한 결과물의 연구논문입니다.

1. 서 론

오늘날 3D 애니메이션 방식은 크게 두 가지로 나누어진다. 하나는 전통적인 방식인 키프레임(Key Frame) 방식이고 나머지 방식은 모션캡처(Motion Capture) 결과물인 모션 데이터들을 비선형(Nonlinear) 방식으로 정리하여 새로운 애니메이션 데이터를 만드는 방식이다. 키프레임 애니메이터가 제작하는 첫째 방식은 시간과 노력이 더 소요되지만 정확한 프레임별 포즈와 타이밍의 창출이 가능하다. 키프레임 애니메이션 방식은 현재도 2D 및 3D 애니메이션을 제작하는 가장 중요한 제작기법이다. 둘째 방식은 기계적으로 적용된 애니메이션 데이터를 기반으로 함으로 높은 효율성을 보장한다. 그러나 콘텐츠 제작에 적극적으로 적용되지 못하는 이유는 타이밍 시트(Timing Sheet)를 기반으로 동작의 자세(Pose)와 적절한 시간성(Timing)이 표현되어야 하는데 연기자의 연기에 의존하는 모션캡처 데이터를 이용한 비선형 방식에 이를 적용하는 것은 접근 방식부터 다르기 때문에 키프레임 애니메이션을 제작하던 경험자들이 모션캡처를 이용한 비선형 방식의 애니메이션 제작을 꺼려온 것이다. 그러나 세계적인 영상전문가들은 보조 캐릭터들이 무리를 지어 움직이는 '반지의 제왕'이나 정교하고 사실적인 움직임을 표현한 '베오울프'와 같은 새로운 영상의 창조에 도전하고, 움직임에 대한 다양한 시각적 표현을 위해 비선형 애니메이션 방식을 적극적으로 활용하며 해결책을 찾고 있다.

1.2 연구목적

모션캡처 데이터는 프레임마다 키프레임이 적용되어 있는 형식인데 기존의 클립형식의 레이어 구성에 의한 수정은 애니메이션 클립 제작 단계 뿐 아니라 키프레임을 클립으로 전환하는 단계 등이 추가되어 키프레임 애니메이터들에게 불편과 부적응을 야기한다. 본고에서는 클립을 적용하여 수정하는 단계 없이 키프레임 애니메이터에게 익숙한 키프레임 수정방식을 사용하게 하여 제작진들에게 가장 익숙한 환경을 제공하여 데이터 수정의 효율성을 얻고자 한다. 또한 일반적인 모션캡처 애니메이션의 수정은 수많은 프레임마다 키를 지정하는 느린 과정이므로 보다 쉬운 방법을 제공하고자 한다.

1.3 연구방법 및 범위

본 연구 방법을 표현하기 위해 두 가지 기술이 필요하다. 첫 번째는 기존의 애니메이션 데이터 값을 기반으로 새로운 키프레임 애니메이션 값과의 데이터간 혼합이 될 수 있는 블랜드 레이어 구축이다. 두 번째는 기존의 값을 무시하고 새로운 애니메이션 값을 가질 수 있는 대치 레이어 구축이다. 손이나 발이 물체에 고정되거나, 걷는 동작에서 땅에 발이 고정되게 하는데 사용할 수 있다. 이 두 가지 모두 기존의 모션캡처 애니메이션 데이터 값과는 독립적으로 구축된다. 서로의 데이터의 훼손 없이 각 레이어가 on/off 되어 기존 데이터에 대해 보존 및 확인하여 새로운 수정을 위한 애니메이션 레이어의 개수와는 상관없이 구축이 가능한 것을 목표로 한다. 그리고 이들은 기존의 애니메이션 데이터와의 직접적인 노드(Node)연결을 하는 레이어 노드 개발로 키 애니메이션 방식을 유지하는 직관적인 방식 구현이 목표다. 구현을 위한 UI는 Maya의 오브젝트 레이어와 렌더 레이어를 구현한 부분에 또 다른 탭으로 구현하여 기존 레이어 컨셉과 일관성을 가질 수 있도록 한다. 이들의 구현은 Maya API를 활용할 것이고 연구범위는 Maya 3D 소프트웨어로 제한하였다.

이를 위해 기존의 애니메이션 방식을 이해하고 이를 기반으로 제작된 상용 애니메이션 툴들을 비교분석하여 모션캡처 데이터 애니메이션을 쉽게 수정할 수 있는 새로운 모델을 제시하고자 한다.

2. 이론적 배경

애니메이션 레이어를 위해 먼저 키프레임 애니메이션과 비선형 애니메이션과의 개념 및 적용의 차이를 이해할 필요가 있다. 또한 애니메이션을 이루기 위해 키프레임 그 자체 외에도 이들 키프레임들을 연결해 주는 인비트윈(In between)과 브레이크다운(Breakdown) 방식이 있는데 이런 연결부분들을 위한 애니메이션 그래프상의 여러 가지 옵션들의 이해를 통해 본 연구의 개념을 설명하고자 한다.

2.1 키프레임 애니메이션과 비선형 애니메이션

2.1.1 키프레임 애니메이션

키 설정은 타이밍(Timing)과 포즈(Pose)를 지정

하는 키를 만드는 과정이다. 애니메이션은 시간의 변화에 따라 오브젝트의 속성(Property)을 만들고 편집하는 과정이다. 키는 특정한 시간에 오브젝트의 속성 값(Value)을 지정하는 일종의 마커(Marker)라고 할 수 있다[1,2].

키 설정은 속성 값을 만들 곳의 시간으로 이동하여, 값을 설정한 다음 그 곳에 키를 넣는 방식으로 이루어진다. 키와 키 사이의 시퀀스(Sequence)는 재배치(Rearrange), 제거(Remove) 및 중복 (Duplicate) 할 수 있다. 컴퓨터는 키와 키 사이의 인비트윈 시퀀스는 보간법을 통해 자동 생성되어 중간 포즈들을 구성하게 된다. 키 사이의 간격이 크면 클수록 인비트윈 포즈들은 어색하게 되고 수정이 필요하게 된다 [3,4].

2.1.2 비선형 애니메이션

비선형 애니메이션 개념(그림 1)인 모션 재 방향 설정(Motion re-direction)은 캐릭터의 경로 또는 방향을 애니메이션 중 언제든지 수정할 수 있는 프로세

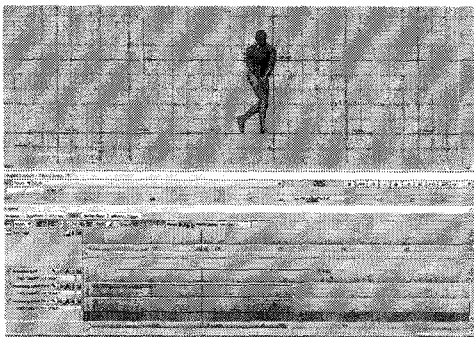


그림 1. 비선형 애니메이션 적용 예

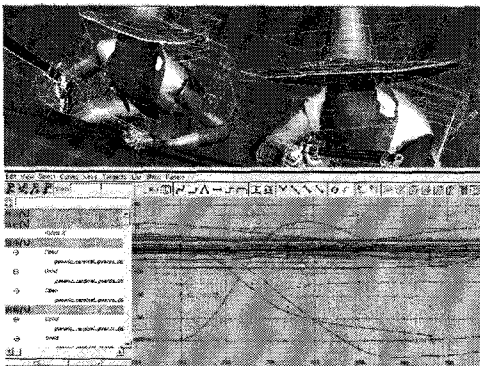


그림 2. 키프레임 애니메이션 적용 예

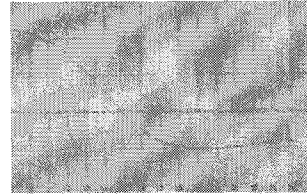


그림 3. 타임 11의 키가 타임 17로 브레이크다운이 없이 변형됨

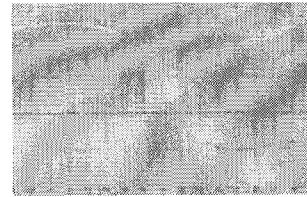


그림 4. 그림 3 에서 브레이크다운이 비례적으로 위치 수정

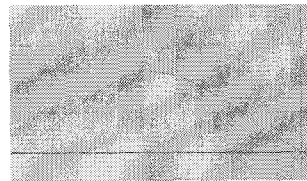


그림 5. Spline 곡선

스를 가리킨다. 예를 들면, 캐릭터의 모션을 재방향 설정하여 해당 캐릭터의 점프 애니메이션 끝에 캐릭터가 착지할 지점과 착지 방향을 변경할 수 있다. 새로운 재 방향설정 컨트롤을 사용하면 애니메이션 중 어느 지점에서나 캐릭터를 이동 및 회전하고 새로운 위치 및 방향을 입력할 수 있다. 예로 걸어가는 동작이나 뛰어가는 동작, 점프하는 동작, 또는 구르는 동작은 비선형 연속 애니메이션으로 만들기에 적합한 동작이다. 이유는 다양한 방법으로 이와 같은 동작을 블렌딩하거나 연속적으로 표현할 수 있기 때문이다. 어떤 군인이 걷고 있는 동작을 행진하는 것으로 보여 지길 원하면 평범한 걸음보다 무릎을 높게 지정해서 마치 그가 행진하는 것처럼 보이길 원할 것이다. 이때 군인 캐릭터의 나머지 부분에는 영향을 끼치지 않으면서 다리의 움직임을 조절하는 키를 더하는 것(그림 2)으로 원하는 효과를 연출할 수 있다[5-7].

2.2 키프레임 간의 구성을 위한 것들

2.2.1 브레이크다운(Breakdowns)(그림3,4)

브레이크다운은 인접 키 와의 비례적 시간 관계를



그림 6. Linear 직선



그림 7. Stepped 선

유지하는 특별한 키다. 브레이크다운은 애니메이션 커브의 한 지점에 속성 값을 저장하면서 애니메이션 타이밍을 맞출 때 사용한다. 인접 키 브레이크다운은 종속되지 않는다. 이것은 브레이크다운을 놓거나 편집, 이동할 때 정규 키와 같은 역할을 한다. 브레이크다운에는 편집할 수 있는 탄젠트커브(Tangent Curve)가 있다. 하나 이상의 브레이크다운이 정규 키에 종속되어 있는 경우, 브레이크다운은 종속된 정규 키의 시간 위치의 변경을 반영하여 시간 내의 위치를 수정한다[8].

2.2.2 인비트윈(In-between)

애니메이션 일루전(Illusion)을 만들어내기 위해 애니메이션의 중심이 되는 키프레임과 이들을 연결하여 주는 인비트윈으로 구성된다. 일반적인 CG 애니메이션에서 인비트윈은 정확하게 컴퓨터가 자동

으로 계산하여 키프레임 간의 중간 포즈를 나타낸다. 효과적인 표현을 위해 타이밍의 조절을 위한 여러 옵션들이 있다. 이들은 키프레임 간의 연결 방식으로 스플라인(Spline)(그림 5)같은 곡선 연결과 리니어(Linear)(그림 6)처럼 직선 연결, 스텝(Stepped)(그림 7)처럼 계단식 연결이 있다. 스플라인(Spline)은 타이밍 조절에 의해 생동감 있는 애니메이션을 만들 수 있고, 리니어(Linear)는 기계적 표현에 적합하고, 스텝(Stepped)은 2D 애니메이션처럼 키프레임과 키프레임 사이에 점진적인 단계가 없다[9].

즉, 인비트윈을 사용하면 프레임이나 시간 단위 하나를 추가하거나 이동하여 애니메이션 타이밍을 조정할 수 있다. 키와 브레이크다운은 자신의 탄젠트 특성을 저장해 놓아 기존 애니메이션 커브 변경을 최소화한다.

2.2.3 상용 툴 간의 애니메이션 적용 비교

기존의 상용 3D 소프트웨어들은 표1에 보는 바와 같이 키프레임 애니메이션과 비선형 애니메이션 어느 한쪽의 기능이 부각되는 현상을 보인다. 한 소프트웨어 안에서 두 가지 애니메이션 제작 방식을 포함하려 시도한 예로써 Maya에는 “Maya Trex”라는 항목을 추가해 비선형 애니메이션 수정을 용이하게 했다. 본 연구에서는 두 가지 수정방안이 해결되도록 하나의 기능으로 합쳐 키프레임 데이터를 클립으로 만들어 비선형 애니메이션을 제작하고 그 클립을 키프레임으로 변형시키는 절차를 가능하게 한다.

3. 모션캡처 데이터 수정을 위한 애니메이션 레이어 방식 제안

두 가지의 모션캡처 데이터 수정을 위한 애니메이

표 1. 제안 방식과 기존 시스템 비교 분석

상용 3D 소프트웨어	키프레임 애니메이션 적용 난이도	비선형 애니메이션 적용 난이도	적합 분야
Maya	상	하	그래프에디터를 이용한 키프레임 애니메이션
3D Studio Max	상	중	타임라인을 이용한 키프레임 애니메이션
Motion Builder	중	상	모션 데이터 클립을 이용한 비선형 애니메이션
Maya Trex	하	상	모션 데이터 클립을 이용한 비선형 애니메이션
Endorphine	하	중	모션 데이터 클립을 이용한 시뮬레이션 애니메이션
제안방식	상	상	비선형 애니메이션 구현으로 수정하기 어려운 모션캡처

선 레이어 처리 방식이 있다. 첫째, 두 가지 모션 데이터 값을 혼합하는 것으로 모션의 자연스러운 합성에 관한 것이다. 둘째, 필림현상이 있는 모션캡처 데이터에 적용될 모션 대치 방식에 관한 것이다.

3.1 블렌드 레이어(Blend Layer)의 컨셉

비선형 애니메이션의 핵심은 기존의 애니메이션의 모션을 그대로 유지하면서 새로운 애니메이션 모션을 부가하여 전체적인 애니메이션의 모션을 만들어가는 방식이다. 기존의 모션을 그대로 유지 하면서 새로운 모션을 부가한다면 포토샵의 레이어 개념과 유사하게 이해해야 한다. 하위의 레이어 모션이 상위의 레이어 모션에 영향을 받아 두 모션의 블렌드된 모션이 최종 산출 된다[10,11].

그림 8은 아래위로 균일하게 반복하는 동작(Original Base Animation)을 나타낸다. 빨간 대각선은 혼합할 동작값을 나타낸 것이다. 그림 9에선 두 가지의 움직임들이 혼합된 애니메이션을 나타낸 것으로 같은 위치값으로 반복된 동작이 위로 향하고 있는 것(Out_Put Animation)을 나타낸다. 즉, 최종결과 애니메이션은 그림 10처럼 기존의 동작 곡선과 블렌드 레이어(Blend Layer)를 합한 값으로 구성된다.

3.1.1 실제 내부에서 필요한 블렌드 레이어의 산술적 계산

실제 애니메이션을 키프레임으로 한다는 것은 애

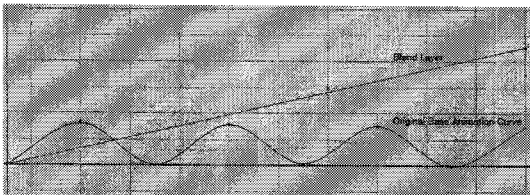


그림 8. 원래 만들어진 기존의 모션 커브(Original Base Animation Curve)와 혼합을 하기 위해 추가한 모션 커브(Blend Layer)

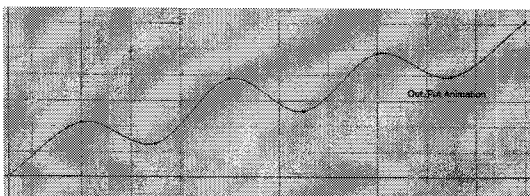


그림 9. 위치 값의 산술적 합에 의한 결과(Out_Put Animation)

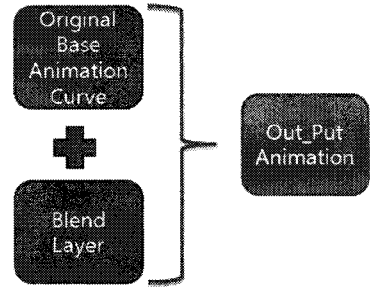


그림 10. 최종결과 애니메이션(Out_Put Animation) 구성

니메이션 대상이 될 오브젝트에 위치값이 최종 계산된 결과를 이른다. 따라서 이값을 기준으로 블렌드 레이어(Blend Layer) 키 값을 역산하여 적용한다. 즉 눈으로 보이는 애니메이션은 최종적으로 혼합된 키프레임의 결과 값이다. 이것은 인위적으로 조절되는 값이지 위의 구성 그림 10처럼 기존의 모션커브와 블렌드 레이어(Blend Layer)의 합으로 계산될 수는 없다. 즉, 역으로 인위적으로 지정된 최종 결과 값에서 기존의 키프레임 곡선 값을 제하면 블렌드 레이어(Blend Layer)의 값을 추출할 수 있다. 그리고 이 블렌드 레이어(Blend Layer)의 값으로 키프레임을 지정하면 자연스럽게 결과 값이 재계산되어 3D 공간상에 표현된다.

그림 11처럼 원래 좌표가 (1,2)일 때 애니메이터가 원하는 좌표 (1,3)까지 움직이게 키프레임을 만들었다면 역으로 블렌드 레이어(Blend Layer)에 해당하는 위치 값이 계산되도록 구성해야 된다. 왜냐하면 애니메이터들이 계산된 위치인 좌표 (1,2)만큼 움직이면 기존의 값이 좌표 (1,1)이니 최종 위치가 좌표 (1,3)으로 될 것이라고 생각하면서 애니메이션을 한

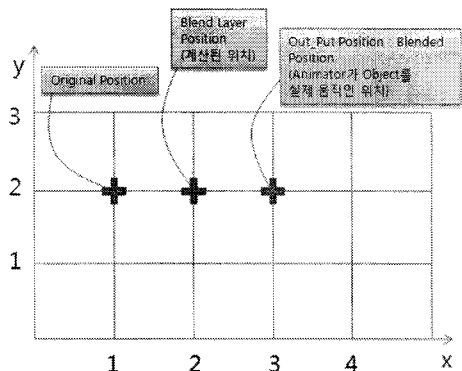


그림 11. 블렌드 레이어의 산술

다면 오히려 많은 수정이 필요한 오류를 발생 시킬 것이기 때문이다.

그러므로 키프레임을 만들 때마다 블렌드 레이어(Blend Layer)값 계산이 필요하다. 계산을 수행하기 위한 흐름은 그림 12와 같다.

그림에서처럼 애니메이터가 블렌드 모션의 결과를 위해 실제 위치를 변화시켜^①, 블렌드 레이어 위치 값이 역으로 계산되고^②, 이 값이 블렌드 레이어 노드 위치에 대입이 된다^③. 다시 최종 얻어질 값을 위해 정상적인 계산으로 기존의 위치와 계산된 블렌드 레이어 위치의 합이 계산되고^④, 마지막으로 최종 계산된 값이 표시 된다^⑤.

애니메이터가 블렌드 레이어 키프레임을 만들기 위해선 초기 애니메이션 레이어 위에 블렌드 레이어(Blend Layer) 설정이 필요하고 그 다음 과정으로 블렌드 레이어(Blend Layer) 계산 과정을 지나 최종 블렌드 레이어(Blend Layer) 키프레임을 만들 수 있다.

실제 노드 구조는 그림 13처럼 블렌드 레이어(Blend Layer)를 만들 때 마다 초기 애니메이션 값을 가지고 있는 노드와 최종 결과를 산출하는 노드 사이에 레이어별로 하나씩 생성되는 구조로 되어 있고 각 노드마다 위 계산과정을 가지고 있다. 최종 결과 애니메이션이 블렌드 레이어(Blend Layer)의 영향으로 값이 조절이 되면 그 이후 프레임의 초기 위치 데이터는 숨겨져서 쉽게 알 수 없다. 그러므로 초기

위치 값(Original Position)을 알 수 있는 함수가 필요하다.

3.1.2 초기 위치값 구함(Get Original Position)

초기 위치값을 알기 위해 초기에 부가된 블렌드 레이어와 대치 레이어를 정지(Disable)로 만든 후 그 값을 추출하면 된다. 즉 초기 애니메이션노드와 최종 노드 사이의 애니메이션 레이어 노드의 계산 함수를 정지시키면 초기 애니메이션 노드와 최종 노드가 직접 연결된 효과가 발생함으로 초기 위치 값을 쉽게 구할 수 있다. 그러나 만약 기존 노드에 하나의 레이어 노드가 아닌 여러 겹의 레이어 노드가 있다면 간단한 문제는 아니다. 현재 선택된 오브젝트와 선택된 레이어에 연계된 노드들을 찾으면 이들이 모두 누적된 레이어들을 포함함으로 이들 모두를 정지(Disable)시키면 된다. 이때 선택된 오브젝트와 선택된 레이어와 연계된 노드들을 찾아내는 함수가 필요하며 레이어를 구성하는 노드들의 네이밍이 중요한데, 위와 같은 노드들의 관계를 이용한 함수들을 만들 때는 노드자체의 이름을 활용함이 효과적인 방법이다[12].

그림 14와 같이 노드자체에 관련정보자체를 이름으로 만들면 정보를 찾기가 쉬워진다.

3.2 대치 레이어(Replace Layer)의 컨셉

일반적으로 애니메이션 레이어는 두 가지 동작 값이 합쳐지는 것을 중심으로 구성하게 된다. 그러나 모션캡처 데이터의 특성상, 즉 일반적으로 모션캡처된 데이터는 약간씩 떨림 현상이 존재한다. 각각의 장비의 특성과 캡처 당시 환경에 따라 이 떨림이 클 수도 있으므로 이를 완화 할 수 있는 도구가 필요하다. 이는 대치 레이어로 보정이 가능하다. 물론 떨림 제거만의 목적으로 대치 레이어가 존재 하는 것은 아니다. 모든 캡처된 데이터가 원하는 대로 될 수는 없다. 때론 부분적으로 완전히 다른 모션으로 대체를

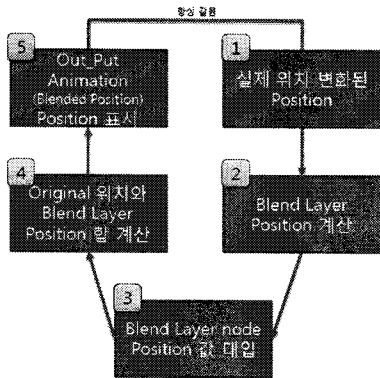


그림 12. 블렌드 레이어 값 계산 흐름

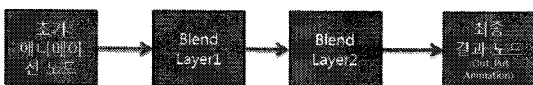


그림 13. 블렌드 레이어 노드 구조

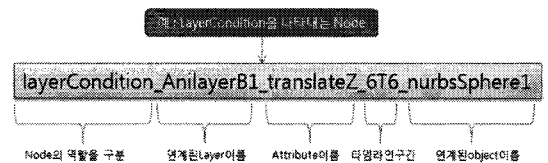


그림 14. 레이어 노드의 네이밍

할 때도 필요하다[13].

대치 레이어에서는 기존의 모션 데이터의 손실이 없이 필요부분을 대체하는 것으로 특히 모션캡처 데이터의 불완전한 모션 오류를 쉽게 수정할 수 있다.

그림 15에서와 같이 대치 레이어를 구현하기 위해 선 적용구간이 필요한데 이유는 타임라인 상에 모든 구간이 대치 레이어의 영향을 받는다면 이 레이어 이하의 레이어는 모두 영향을 받음으로 다른 레이어들이 쓸모없게 된다. 그런 이유로 구간적용이 필요하다. 그래서 설정된 구간에서는 그 이하의 레이어는 영향을 받음으로 새로운 레이어로 대치가 된다. 그러나 3D 공간 3가지 축 x, y, z는 서로 독립적이므로 대치 레이어도 독립적으로 적용이 된다.

그림 16처럼 대치 레이어는 대치 ON/OFF 노드의 구성에 의해 ON/OFF가 되어 기존 레이어를 대체할 수 있다. 비주얼 ON/OFF 노드는 최종 결과의 모습을 모여 줄 수 있는 구성이다.

적용의 예로 주로 모션캡처 데이터의 떨림 현상의 보정이나 애니메이션 정지를 걸어 줄때 유용하게 사용될 수 있다. 또는 모션캡처 데이터의 실사적인 모션의 유형을 줄일 때 사용될 수 있는데 처음과 끝의 구간을 설정하고 애니메이션 일루전을 나타낼 수 있는 중요키만 대치 레이어에서 남겨 그림 17처럼 실사 동작의 흔적을 지울 수 있다. 이는 모션캡처 데이터를 카툰식 캐릭터에 적용할 때 매우 유용하게 적용될 것으로 본다.

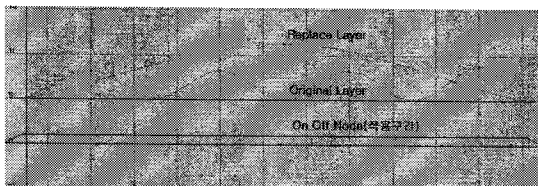


그림 15. 대치 레이어 그래프

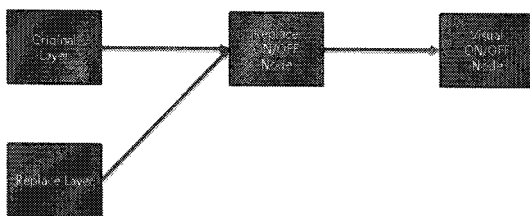


그림 16. 대치 레이어 구성

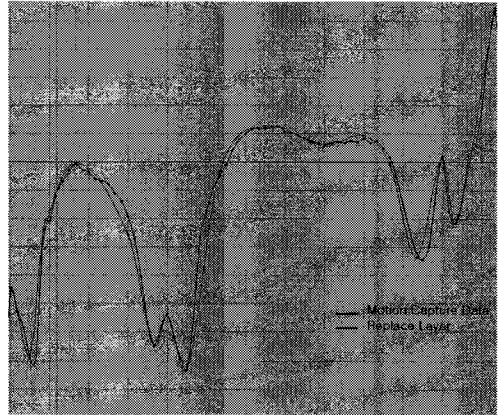


그림 17. 실사모션의 보정

3.3 애니메이션 레이어의 구성

3.3.1 애니메이션 레이어(UI 레이어, 노드 레이어, 오브젝트)

애니메이션 레이어가 존재하는 파일을 다시 열 때 UI로 레이어 정보를 어떻게 재구성하는가라는 문제가 있다. 이 문제는 레이어 노드 생성 시 이미 네이밍으로 정보를 노드 자체에 포함함으로써 해결할 수 있다. “pair블렌드_”로 시작되는 노드는 대치 레이어나 블렌드 레이어 공동으로 가지고 있도록 한다.

그러므로 모두 “pair블렌드_” 노드를 찾으면 연계된 레이어 이름을 추출할 수 있다.

그림 18의 구성 방식은 결국 UI 레이어(사용자가 선택할 수 있는 표면적 레이어)와 노드 레이어(내부적으로 실제 레이어를 이루는 노드들)의 연계를 구성할 수 있는 함수 기초가 된다.

정보를 추출하기 위해 그림 19와 같이 3 구성요소의 연계가 필요하다. 특정 오브젝트는 특정 레이어에 속하고 그 레이어는 특정 노드를 만들기 때문에 레이어 생성 시 3가지는 항상 같이 생성과 소멸을 하도록 한다. 이런 함수로 이 3가지 중 어느 하나를 알면 나머지 정보를 추출할 수 있도록 함수를 구성한다. 이

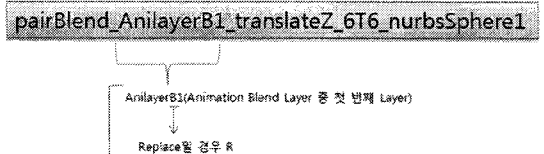


그림 18. 애니메이션 레이어 이름 구성

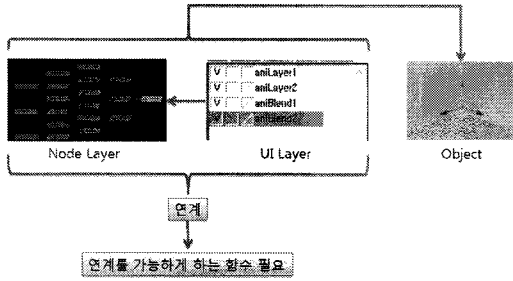


그림 19. 노드 레이어, UI 레이어, 오브젝트 구성

함수는 그림 14에서 설명한 노드 네이밍을 기초로 구성된다.

3.3.2 레이어 지움(Delete Layer)

그림 20처럼 본 함수의 활용으로 레이어의 지움에 활용할 수 있는데, 레이어를 지운다는 의미는 단순히 UI 레이어를 지운다는 것이 아니라 내부적으로 연계된 노드들을 찾아 같이 지운다는 의미임으로 연계된 노드들의 지움이 함수의 활용으로 가능하다.

3.3.3 머지 레이어(Merge Layer, 레이어 통합)

그림 21은 기존의 레이어에 만들어진 블렌드 레이어

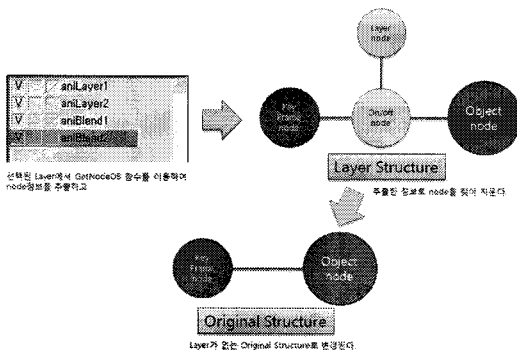


그림 20. 레이어 지움 과정

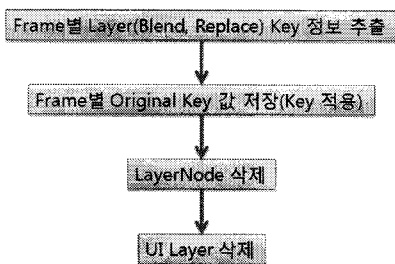


그림 21. 머지 레이어 흐름

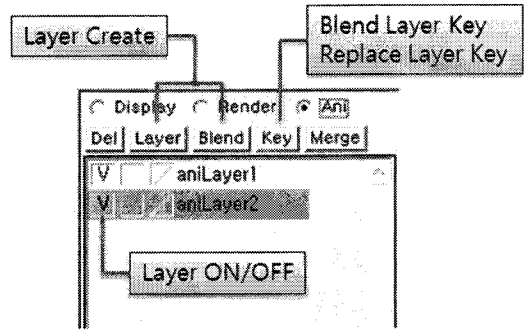


그림 22. UI 구성

이나 대치 레이어의 값을 모두 합하여 적용하는 레이어 통합의 흐름을 말한다. 그리고 기존의 UI 레이어나 노드 레이어는 삭제하고 단일 레이어에 합한 키 값을 적용한다.

3.3.4 UI 구성

그림 22에서 보듯이 애니메이션 레이어의 UI 구성은 기존의 디스플레이 레이어와 렌더 레이어가 있는 "Channel Box" 밑 부분에 위치하여 기존의 레이어 UI에 연속성을 유지하였다.

4. 제안방식과 비교

4.1 기대효과

애니메이터에 의한 키프레임 애니메이션으로 제작된 애니메이션은 재능이 많은 인력의 구축과 기간이 필수적으로 소요되고 그에 상응하는 제작비가 필요하다. 우리나라 보다 애니메이션 제작 여건이 좋은 일본의 애니메이션 산업조차 이러한 구조를 갖추지 못하고 있다. 일본 애니메이션 제작사들 중 대표적인 '곤조' 스튜디오 역시 적은 인원의 3D 도구를 다루는 제작진이 애니메이션에 들어가는 CG를 도맡아 하고 있는 것이 현실이다. 모션 캡처를 활용한 애니메이션 제작의 요청이 높아지게 되지만, 활용에 부정적인 요소들로 전통 애니메이션과의 접근방식의 차이와 데이터 수정의 어려움이 작용하였다. 따라서 클립을 이용하거나, 모션캡처가 적용된 캐릭터 뼈대에 IK (Inverse Kinematics)를 이용하여 수정하고 있다. 본 연구에서 밝힌 도구 적용과 일반적인 모션캡처 데이터 수정 작업과 비교를 하면 아래 표와 같다. 기준은 기본 운동 자세인 걷기동작에 대한 모션캡처 데이터

표 2. 제안 방식과 기존 방식 비교 분석

모션캡처 수정 시 비교	전통적인 단순 키프레임 수정 방식	Maya Trax를 활용한 방식 (Clip이용)	제시한 애니메이션 레이어 Tool 적용
모션캡처 데이터 선행 작업 불필요	○ (필요 없음)	× (블렌드 할 소규모 단위 마다 Clip으로 전환)	○ (필요 없음)
기존의 데이터 수정 불필요	× (직접적인 키프레임 수정 필요하고 이 경우 원본 값 유지가 어려움)	○ (필요 없음 : clip으로 만들기 위한 선행 작업 소요)	○ (필요 없음)
간단한 수정 작업	× (프레임 별 키 수정으로 매우 복잡함)	○ (Clip 레이어로 수정)	○ (직접 키프레임과 연계된 노드 레이어 수정)
떨림 등의 오류 모션 수정	× (매우 어려움)	× (레이어 방식으로 수정 어려움)	○ (대치 레이어로 쉽게 가능함)
메인 키 Pose 추출	× (매우 어려움)	× (매우 어려움)	○ (가능)
Interface	× (직관적이나 수행이 어려움)	× (직관성이 떨어짐)	○ (매우 직관적임)
타 3D 프로그램과의 애니메이션 데이터의 호환	○ (가능)	× (어려움)	○ (가능 : 키프레임 애니메이션 데이터 그 자체 노드에 Merge됨으로 가능)

표 3. 소요일수에 따른 효율성 비교

	전통적인 단순 키프레임 수정 방식	Maya Trax를 활용한 방식	제시한 애니메이션 레이어 Tool 적용
모션캡처	필요 없음	5시간	5시간
수정작업	약 20일	모션캡처 데이터 선행작업(데이터클리닝)-1일 떨림등 모션 오류수정-4일 본 수정 - 9일	떨림 수정-1일 본 수정 - 5일
총 소요일수	약 20일	약 15일	약 7일

수정 작업을 표2로 비교를 하였다.

표 3은 (주)팜킨스튜디오에서 2008년 11월 실시한 연구 결과로 본 연구에서 제시된 방식이 50% 이상의 효율성이 있다는 것을 잘 나타낸다. 부여된 조건은 3년차 애니메이터 2명이 사람 형태 캐릭터 5개의 동작에 대해 타이밍시트로 제시된 총 3분의 애니메이션을 하는 것이었다. 레이어 틀은 수정작업에 있어 기존 키프레임 애니메이션 제작 방식과 유사하여 적용방식이 간단하고, 레이어를 통한 애니메이션 데이터들의 보존 및 수정이 쉬워 애니메이션 제작사들의 제작 공정에 효율성을 높이는데 기여한다.

5. 결 론

본 연구는 모션캡처 데이터를 다루는 과정에서 빈번하게 발생하는 수정의 문제점을 효율적으로 해결하기 위한 애니메이션 레이어 개발에 관한 연구이다. 중점 연구문 두 가지가 있다. 첫째는 기존의 모션캡

처 데이터를 직접적으로 수정하지 않고 레이어를 활용하여 모션캡처 데이터를 자유롭게 수정하는 것이고, 둘째는 떨림 현상이나 기존의 실사 모션의 느낌을 보완하거나 축소할 때 활용되는 방식에 대한 연구이다. 본 연구에서 기초적인 모션 데이터 즉 키프레임을 직접 다룬다는 것은 매우 어려운 일이다. 특히 매 프레임마다 키가 들어 있는 모션캡처 데이터에서 다루어야 하는 키프레임이 매우 많음으로 거의 불가능에 가깝다. 본 연구는 모션캡처 데이터 자체를 다루지 않고 그 모션캡처 데이터와 직접적인 노드로 연결된 레이어를 생성하여 애니메이션 데이터를 수정함으로써 작업의 효율을 극대화 할 뿐 아니라 작업 인터페이스도 매우 간결하여 기존의 애니메이션 작업과 크게 다를 바가 없음으로 보다 직관적으로 작업을 수행 할 수 있다.

제안 방식의 주요 효과로는 모션캡처 데이터의 선행 작업의 불필요, 기존 데이터의 수정 불필요, 비선형 애니메이션 수정방식을 키프레임 방식에 적용하

여 용이한 수정작업, 떨림 등의 모션캡처 데이터의 수정 가능, 사실적 모션을 비사실적 모션으로 전환할 메인 키 포즈 추출 가능, 타 프로그램과 데이터의 호환가능 등이 있다.

결론적으로 애니메이션 수정작업시 모션캡처 데이터와 직접적인 노드로 연결된 레이어를 활용하는 것이 동일하게 제시된 작업량 조건에서 비교한 결과 기존의 방식 대비 50%이상의 효율적인 방법이다.

참고 문헌

[1] Alias, "The Art of Maya," *Alias System*, San Francisco, 2005.

[2] Brad Clark, John Hood and Joe Harkins, "3D Advanced Rigging and Deformations," *Thomson*, Boston, 2005.

[3] 채일진, "애니메이션 효율적 공정관리를 위한 캐릭터 셋업," *한국콘텐츠학회논문지*, 제 7권, 제 4호, pp. 133-140, 2007.

[4] Alias, "Maya 8.5 Help Documents," *Alias System*, San Francisco, 2007.

[5] 김기홍, "다수의 대용량인 3D Render 파일을 위한 3D CG 소프트웨어와 독립적으로 Render Option 값 추출과 적용 가능한 Net-Work Rendering Tool 구현," *한국애니메이션학회논문지*, 제 3권, 제 1호, pp. 7-30, 2007.

[6] 최철영, "효율적인 모션캡처 Animation 수정을 위한 FK Animation을 IK Animation방식으로서의 수정에 관한 연구," *한국디지털영상학회논문지*, 제 3권, 제 2호, pp.151-174, 2007.

[7] <http://www.mone.co.kr/product/renderfarm.asp>

[8] R. Stuart Ferguson, "Practical Algorithms for 3D Computer Graphics," *A.K. Peters*, Wellesley, 2001.

[9] <http://usa.autodesk.com/adsk/servlet/ps/item?siteID=123112&id=5581724&linkID=9241177>.

[10] 최철영, "3D 애니메이션에서 FK애니메이션에서 일반적인 애니메이션 방식인 IK애니메이션으로 전환에 관한 연구," *한국애니메이션학회 논문지*, 제 3권, 제 1호, pp. 109-126, 2007.

[11] 이인호, 박찬중, "모션캡처 기술의 현황과 응용 분야," *한국멀티미디어학회지*, 제3권, 제1호, pp. 38-45, 1999.

[12] David A.D. Gould, "Complete Maya Programming," *Morgan Kaufmann Publishers*, San Francisco, 2003.

[13] 최태준, 유석호, 이동열, 이완복 "광학식 모션캡처(Optical Motion Capture)방식을 이용한 디지털 캐릭터 움직임," *한국콘텐츠학회지*, 제 7권, 제 8호, pp. 112-115, 2007.



김 기 홍

1995년 홍익대학교 화학공학과 공학사
2001년 CalArts Experimental Animation MFA
2002년~현재 동서대학교 애니메이션학과 교수

관심분야 : 컴퓨터그래픽, IT, 문화 콘텐츠



최 철 영

1997년 한림대학교 물리학과 학사
2002년 Academy of Art Univ. Computer Arts MFA
2002년~2004년 (주)디지스팟 애니메이션 팀장
2004년~현재 동서대학교 디지털콘텐츠학부 애니메이션전공 교수

관심분야 : 3D애니메이션, 모션캡처, 비주얼 이펙트, 문화 콘텐츠



채 일 진

1998년 중앙대학교 사진학과 학사
2001년 Academy of Art Univ. Motion Picture & Television MFA
2003년~2004년 (주)디씨디코리아 대표이사
2004년~현재 동서대학교 디지털콘텐츠학부 디지털영상제작전공 교수

관심분야 : 문화 콘텐츠, 기획, 마케팅, 제작관리