

아바타 위치 동기화를 위한 이동 벡터를 이용한 경로 재설정 방법

김종석*, 김학근, 최윤철
연세대학교 컴퓨터과학과

Rerouting Technique with Walking Vector for Avatar Location Synchronization in CVE

Jongseok Kim, Haken Kim, Yoonchul Choy
Dept. of Computer Science, Yonsei Univ.

E-mail : jongdori@hotmail.com, air153@rainbow.yonsei.ac.kr, ycchoy@rainbow.yonsei.ac.kr

요 약

본 논문에서는 다중 사용자 간의 상호 작용과 공동 작업을 가능하게 하는 3차원 그래픽 기반의 가상 현실 시스템인 YVES(Yonsei Virtual Environment System)의 성능향상을 위한 아바타 객체의 효과적인 동기화 방법에 대하여 제안하였다. YVES는 연세대학교 컴퓨터과학과 멀티미디어 연구실에서 2001년도에 개발된 시스템이다.

YVES의 사용자간 동기화 알고리즘은 기존의 방법에 비해 객체의 Popping 현상을 없애고 동기화를 유지한 점에서 개선되었다고 할 수 있었다. 그러나 원래의 경로와는 변화된 다른 경로가 설정된다는 것과 이동이 완료되었을 때 수정된 경로로 향한 객체 진행 방향을 완료시점에서 원래의 진행방향으로 수정해야 함으로써 자연스럽게 못한 움직임을 보인다는 것이 문제점으로 지적되었다. 이를 보완하기 위해서 본 논문에서는 기존의 움직임 벡터를 새로 움직여야 할 움직임 벡터에 반영하여 새로운 경로가 원래의 경로와 보다 근접할 수 있도록 하고, 이동의 완료시점에서 객체가 향하는 방향이 원래 경로로 이동했을 경우와 동일하도록 하는 알고리즘을 제안하고, 각 알고리즘을 동일한 이동경로를 적용하여 비교하였다.

1. 서론

최근의 컴퓨터 하드웨어 및 소프트웨어의 발전은 게임을 비롯한 그래픽 응용 분야에서 3D 그래픽의 사용을 보편화 시키고 있다. 따라서 사용자들은

기존의 2D 그래픽보다 현실감 높은 3D 그래픽 프로그램의 개발을 요구하고 있다. 또한 네트워크 인프라의 확충으로 인터넷 기반의 다중 사용자가 동시에 참여할 수 있는 시스템이 널리 이용되고 있다. 이러한 특징을 고려해 볼 때, 3D 가상 공간을 다중사용자

가 동시에 참여가능하고, 그 안에서 전개되는 상황이 다수 사용자의 현실감과 몰입감을 증가시키는 환경을 개발하는 것은 컴퓨터 가상환경 분야의 중요한 과제라 할 수 있다.

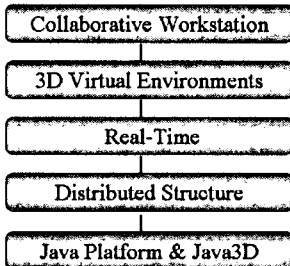
연세대학교 컴퓨터과학과 멀티미디어 연구실에서는 인터넷 기반의 실시간 다중 사용자 가상환경 시스템에 대한 연구를 수행하고 있다. 2001년도에 개발한 YVES (Yonsei Virtual Environment System)은 이 연구에 대한 테스트엔진으로써 여러가지 시험적인 알고리즘들을 포함하고 있다. 이 중에서 아바타간 동기화 문제를 해결하기 위한 방법으로 경로재설정 방법을 연구하였다. 이 방법은 객체의 Popping 현상을 없애고 동기화를 유지했다는 점에서 기존의 방법보다 개선된 점을 찾을 수 있었다.

본 연구에서는 동기화를 유지하는 과정에서 원래의 경로와는 다른 변화된 경로가 설정되고, 이동 완료지점에서 객체 진행방향을 갑작스럽게 변화시켜 주는 부자연스러움을 해결하기 위한 방법을 제안하고자 했다.

제안한 백터기반 경로재설정 알고리즘의 비교를 위해서 기본적인 애니메이션 우선순위 알고리즘과 동기화 우선순위 알고리즘, 그리고 YVES에서 사용되었던 알고리즘을 동일한 이동경로를 설정하여 비교하였다.

2. YVES (Yonsei Virtual Environment System)

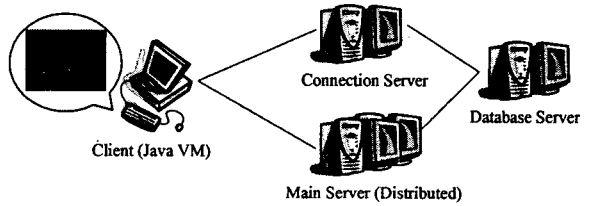
YVES은 다중 사용자 간의 상호작용을 지원하기 위한 엔진으로 2001년 연세대학교 컴퓨터과학과 멀티미디어 연구실에서 개발되었다. 이 시스템은 [그림 1]의 5가지 특성을 가진다.



[그림 1] 시스템의 특성

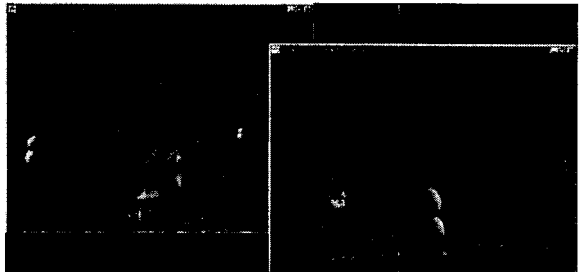
YVES은 MS Window2000 환경에서 JDK 1.3.1, Java3D 1.2.1 beta를 이용하여 구현하였다. 데이터베이스로는 ODBC-MS Access를 사용하였고, Java와 데이터베이스와의 연동을 위하여 JDK에서 제공되는 JDBC-ODBC Bridge 드라이버를 사용하였다. 3D 월드 상에서 다양한 Object를 표현하기 위하여 Java3D에서 제공하는 프리미티브 기하 도형 클래스 이외에 3DS, OBJ, WRL 파일들을 추가적으로 사용하여 현실감 있는 Object를 표현했다. 이와 같은 파일들을 Java3D에서 로딩할 수 있도록 NCSA 파일 로더를 이용하였다.

시스템의 구조는 [그림2]에서 보는 바와 같이 Database Server, Connection Server, Main Server, Client로 구성되어 있으며 클라이언트-서버 모델을 기본으로 인증/보안, 분산 처리, 데이터베이스, 객체와 지역관리, 전송 데이터 압축, 동적인 Scene Graph 구성 등 여러 가지 세부 모듈이 복합적으로 사용 되었다



[그림2] 전체 시스템의 구성

사용자가 시스템에 접속하게 되면 3D 디스플레이 이 윈도우 상에서 사용자는 마우스나 키보드로 자신의 아바타를 조작할 수 있고, 3D 가상 공간을 자유롭게 네비게이션 할 수 있다. 또한 다른 사용자의 움직임이나 기타 동적인 Object들의 움직임을 실시간으로 관찰할 수 있다.

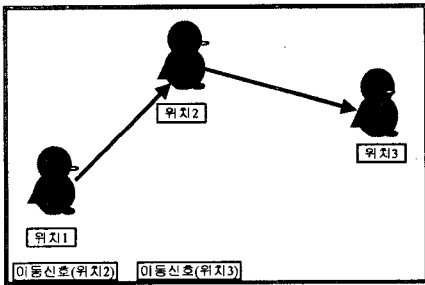


[그림3] 3D 디스플레이 윈도우

3. 아바타 객체 위치 동기화

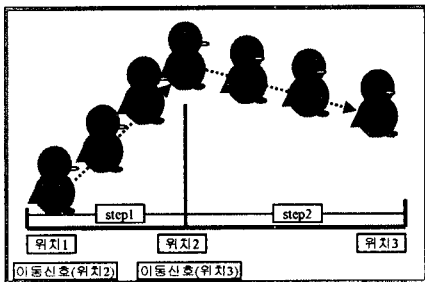
3.1 다른 사용자 아바타의 이동

가상환경 상에서 사용자는 자신의 아바타를 원하는 장소로 이동하게 된다. 그리고 이 움직임은 다른 사용자에게 보여지게 된다. 이를 위해서는 움직임을 관찰 할 사용자가 서버로부터 다른 아바타 이동 데이터를 수신하여야 하고, 이 정보를 이용하여 관찰자 화면의 아바타 이동이 시작된다. 이상적인 이동 예로는 [그림4]와 같이 위치1일때 이동신호(위치2)를 받고 위치2로 이동시킨 후, 다시 이동신호(위치3)을 받고 위치3으로 이동시키는 것이다.



[그림 4] 다른 사용자 아바타의 이동 예

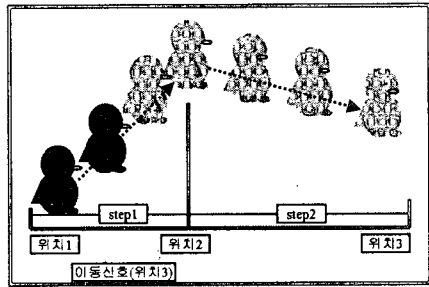
그러나 위치1에서 위치2로 이동을 시킬 경우, 갑작스런 위치 변동은 사용자에게 혼란을 야기할 수 있으므로 [그림5]와 같이 중간단계에 애니메이션을 사용하여 이를 해결한다. 여기서 Step간의 시간은 동일하다.



[그림 5] 중간 애니메이션을 이용한 아바타의 이동

3.2 사용자간의 상이한 클라이언트 환경

네트워크 어플리케이션에서 중요한 고려 사항중의 하나는 동기화를 이루는 것이다. 각 클라이언트는 처리능력이나 회선속도가 다를 수 있으므로, 이에 따라 동기화가 이루어지지 않을 수 있다.

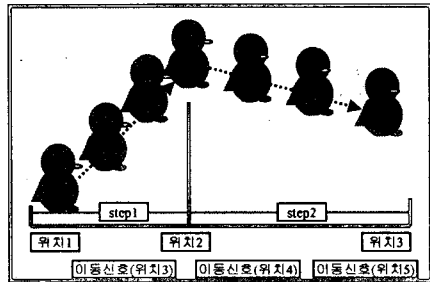


[그림 6] 애니메이션 도중 이동신호가 도착한 경우

[그림6]에서는 애니메이션을 사용하여 이동하는 Step1도중에 더 빠른 시스템으로부터의 이동신호(위치3)가 수신된 경우를 보여주고 있다. 이 경우 처리 방법은 대표적으로 다음의 두가지가 있다.

3.2.1 우선순위를 애니메이션에 두는 경우

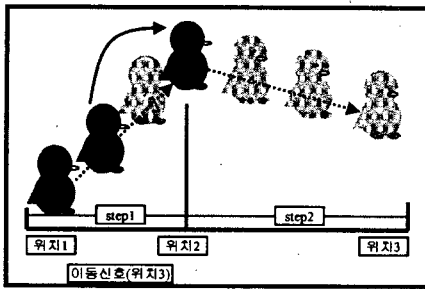
이 방법은 애니메이션을 끝까지 제대로 표현하는 방법이다. 화면의 깨짐 현상이 없으므로 좋은 애니메이션 퀄리티를 얻을 수 있다. 그러나 동기화가 이루어지지 않고, 계속해서 이동신호가 수신될 경우, 이를 누적 처리해야 하는 문제점이 있다. 이동경로나 애니메이션이 중요한 의미를 가지는 경우에는 이 방식을 사용할 필요가 있다.



[그림 7] 애니메이션의 완전한 표현

3.2.2 우선순위를 동기화에 두는 경우

이 방법은 애니메이션을 끝까지 표현하지 않고 생략하여 이동시키는 방법이다. 이 방법은 동기화 문제는 거의 발생하지 않으나 Popping현상(물체가 순간적으로 다른곳에 나타나는 현상)이 발생하여 화면의 움직임 퀄리티를 떨어뜨리고 또한 사용자의 혼란을 야기할 수 있다. 현재의 많은 시스템들이 이 방법을 따르고 있다.



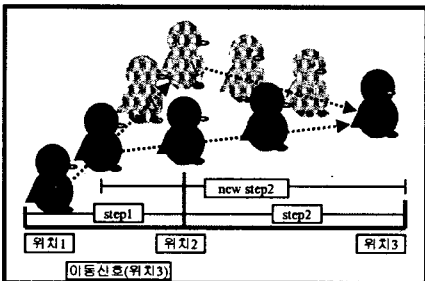
[그림 8] 다음위치로 바로 이동

4. 벡터를 이용한 경로 재설정 방법

4.1 YVES에서의 경로 재설정 방법

YVES에서는 아바타 동기화의 목표로 두 가지를 설정하였다.

- ① 객체의 Popping 현상을 나타나지 않도록 한다.
- ② 객체의 동기화가 이루어 지도록 한다.



[그림 9] 새로운 경로를 설정하여 단위시간에 처리

이것을 해결하기 위한 방법으로 이동신호의 도착 시점에서 최종 도착 지점까지의 새로운 경로를 설정하여 새로운 Step으로 처리한다. 이 새로운 Step은 다른 단일 Step과 동일한 시간으로 처리된다. [그림9]에서는 이 과정을 보여주고 있다.

4.2 벡터를 이용한 경로 재설정 방법

본 논문에서 제안하는 알고리즘은 진행중인 이동 벡터와 새로 진행해야 하는 이동 벡터를 연산하여 보다 개선된 경로를 산출해 내는 알고리즘이다. 애니메이션의 Step수 만큼 이동 위치를 나누고 각 이동 위치는 진행중인 이동벡터와 새로 진행할 이동벡터와의 합으로 구해지는데, 이동 초기에는 기존의 이동벡터로부터 영향을 많이 받고, 후기에는 새 이동벡터로부터 영향을 많이 받게 되는 알고리즘이다.

* 단위 Step내의 총 애니메이션 수 : total

* 현재 진행중인 위치 이동

시작위치 : V_{LOC1} , 도착위치 : V_{LOC2}

현재위치 : V_{LOCCUR}

방향벡터 $V_{first} = V_{LOC2} - V_{LOCCUR}$

* 새로받은 신호

시작위치 : V_{LOC2} , 도착위치 : V_{LOC3}

방향벡터 $V_{second} = V_{LOC3} - V_{LOC2}$

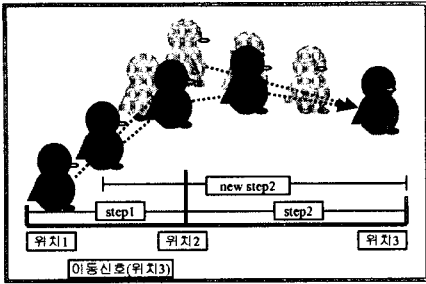
* 새 경로의 n번째 애니메이션에서의 위치와 방향

$V_0 = V_{LOCCUR}$

$V_n = V_{n-1}$

$$+ V_{first} * \frac{total - n}{total} * \frac{2}{total - 1}$$

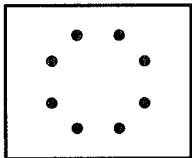
$$+ V_{second} * \frac{n}{total} * \frac{2}{total - 1}$$



[그림 10] 벡터를 이용한 경로 재설정

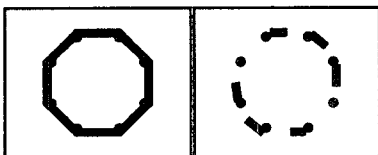
5. 성능 비교

각 알고리즘의 성능을 비교하기 위해서 [그림 11] 과 같은 8개의 위치를 이동경로로 설정하고 임의로 전송지연을 가정하였다.



[그림 11] 성능비교를 위한 이동경로

[그림 12]에서는 기본 알고리즘인 애니메이션 우선순위방법과 동기화 우선순위 방법을 보여주고 있는데 애니메이션 우선순위 방법은 완전한 객체의 움직임을 보여준다. 단 이 경우, 동기화는 이루어 지지 않은 상태이다. 동기화 우선순위 방법은 화면 내에서 객체가 여기저기서 나타나 혼란스러운 화면을 보였다.

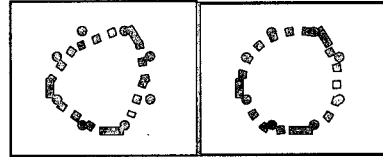


[그림 12] 두개의 기본 알고리즘 이동 경로

(좌측 애니메이션 우선순위, 우측 동기화 우선순위)

[그림 13]에서는 경로를 재설정하는 알고리즘을 보여 주고 있는데 기존의 YVES에서 사용된 새로운 이동

경로 추가 시 즉시 새로운 목표로 향하는 알고리즘은 원래 경로로부터 왜곡이 발생한 것을 볼 수 있다. 본 논문에서 제안한 벡터기반 알고리즘은 왜곡이 발생하였으나 그 오차가 적은 것을 볼 수 있다.



[그림 13] 경로 재설정 알고리즘 이동 경로

(좌측 기존 YVES알고리즘, 우측 벡터기반 알고리즘)

6. 결론

본 연구에서는 방향벡터를 이용하여 보다 개선된 경로를 얻을 수 있는 알고리즘을 제안하였다. 성능비교 결과 이 방법은 객체의 Popping현상을 없애면서도 원래의 경로에 보다 근접한 움직임을 보여줄 수 있었다.

본 방법은 Popping현상을 완전히 없애는 것이 아니라 애니메이션의 중간중간에 아주 미소한 간격을 동으로써 해결하는 것이다. 움직이는 객체이므로 이는 사람이 인지하기 어렵기 때문에 좋은 해결책이라 할 수 있다.

[참고문헌]

- [1] 김종석, 이진상, 최윤철, “다중사용자의 공동작업을 위한 3차원 가상환경 기반 설계 및 구현”, 멀티미디어 춘계학술대회, 4권2호, p441-446, 2001
- [2] 김종석, 김중석, 최윤철, “가상환경에서 아바타 사 고정보의 표현방법”, HCI 2002학술대회, 2002 [1] D. Ball, M. Mirmehdi, A Prototype Hotel Browsing System Using Java3D, Proceedings of the 1999 International Conference on Information Visualisation, IEEE, 1999.
- [3] M.C. Schraefel, Janet Ho and Mark Chignell, Michael Milton, Building Virtual Communities for Research Collaboration, Proceedings of the Academia/Industry Working Conference on Research Challenges

(AIWORC'00), IEEE, 2000.

- [4] Runhe Huang and Jianhua Ma, A General Purpose Virtual Collaboration Room, Proceedings of the 5th International Conference on Engineering of Complex Computer Systems, IEEE, 1998.
- [5] M.C. Schraefel, Janet Ho and Mark Chignell, Michael Milton. , Building Virtual Communities for Research Collaboration, Proceedings of the Acadmia/Industry Working Conference on Research Challenges (AIWORC'00), IEEE, 2000.
- [6] John Vince, Rae Earnshaw. , Virtual world on the internet, book, 1999.
- [7] Shervin Shirmohammadi, Nicolas D. Georganas. , An Architecture for Collaboration in Virtual Environments, Proceedings of the IEEE Virtual Reality 2000 Conference, IEEE.
- [8] Jon Barrilleaux, 3D User Interfaces with JAVA3D, Book of Manning, 2000.