

온라인게임 콘텐츠기술

임 충 규, 이 현 주(한국전자통신연구소)

차 례

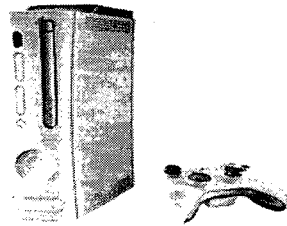
- I. 서론
- II. 게임 클라이언트 기술
- III. 게임 서버 기술
- IV. 게임 인공지능 기술
- V. 결론

I. 서론

최근 몇 년 동안 컴퓨터 게임은 세계적으로 눈부신 성장을 거듭하고 있다. 이미 매출액 면에서 영화관의 박스 오피스(Box Office)의 규모를 추월하였으며, 전문가들은 향후 3,4년 이내에 그 동안 가장 대표적인 오락 거리인 TV를 컴퓨터 게임이 대체하리라 예상하고 있다.

국내에는 97년 이후 PC방이 급속히 보급되면서 온라인 게임이 급격한 성장을 하였고 그 동안 국내 게임 산업의 성장을 이끌고 있다. 이러한 경향은 당분간 지속되리라 예상되지만 2002년 주요 콘솔게임기의 국내 출시로 국내 게임 산업도 게임플랫폼 관점에서 다양하게 발전하고 있다. 2005년에는 대표적인 게임 콘솔인 Playstation2와 Xbox이 차기 버전인 Playstation3와 Xbox360의 출시가 예상되고, 이에 따라 전 세계적으로 게임 산업은 비약적인 성장을 하리라 예상된다. 이러한 플랫폼 외에도 다수의 하드웨어 플랫폼이 출시되고 있는데, 최근에 출시되는 게임 플랫폼은 온라인 기능을 지원하거나 확장하고 있다. 이러한 온라인화 경향은 향후 컴퓨터 게임은 온라인 게임이 주류가 되리라 예상해볼 수 있다.

게임 산업의 성장과 함께 게임 기술은 그 동안 많은 발전을 이루어 냈고 특히 게임 클라이언트 기술의 발전은 컴퓨터 게임의 가상 공간이 현실감을 증대시킬 수 있는 토대가 되고 있다. 이는 컴퓨터 게임 분야 이외에도 사이버 박물관, 가상 쇼핑몰 등과 같이 현실감이 중요한 분야에서 광범위하게 사용될 수 있다. 본 글에서는 컴퓨터 게임의 최신 기술을 살펴봄으로써 실감형 미디어 기술로서 컴퓨터 게임 기술의 응용 가능성과 향후 발전 가능성을 고찰한다.



▶▶ 그림 1. Xbox360의 외관



▶▶ 그림 2. PS3의 디자인 컨셉

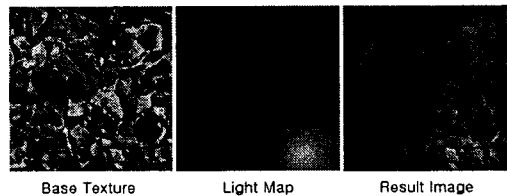
II. 게임 클라이언트 기술

3차원 그래픽을 활용하는 게임은 지난 십여 년 간 눈부신 발전을 하였는데 크게 두 가지로 요약할 수 있다. 하나는 그래픽 가속기의 등장과 일반화이다. 예전에는 며칠씩 걸리던 컴퓨터 그래픽 연산을 실시간으로 처리해주는 그래픽 가속기의 등장하게 되었고 이러한 가속기는 PC 게임 산업의 발전에 따라 저가의 PC에서도 쉽게 찾아볼 수 있게 되었다. 또 하나의 특징은 그 동안 3D 애니메이션 영화의 렌더링 기술인 OpenGL API(Application Programming Interface)를 대신하여 실시간 렌더링 기능이 대폭 보강된 DirectX API의 등장이다. 이는 PC 게임의 그래픽 수준의 발전에 기여하였고 컴퓨터의 계산 능력과 함께 보다 풍부한 게임플레이의 게임 개발이 가능하게 되었다. 본 장에서는 게임 클라이언트의 대표적인 기술인 렌더링 기술, 애니메이션 기술, 사운드 기술 중에서 핵심적인 기술을 열거하고 이들의 기술 동향을 살펴보고자 한다.

2.1 렌더링 기술

그래픽 가속기의 등장으로 이제 많은 컴퓨터 그래픽 연산은 실시간 처리가 가능해졌다. 이러한 가속기의 등장은 TnL(Transform and Lighting)과 Rasterization을 별도의 소프트웨어가 처리하지 않고 그래픽 가속기가 처리함으로써 실시간 처리가 가능하다. 그럼에도 불구하고 빛과 그림자의 처리는 아직 많은 계산량이 소요되는 연산으로 별도의 처리 방법인 다중 텍스처 기법이 개발되어 활용되고 있다. 빛의 처리는 일반적으로 많은 계산이 필요한 그래픽 연산이고 특히 이중에서 Spotlight의 처리는 많은 연산이 필요로 한다. [그림 3]과 같이 밝은 영역에 해당

하는 2차원 이미지를 준비하고 이를 광원의 이동과 환경의 변화에 따라 렌더링하는 기법이다. 다중 텍스처 기법은 그림자의 표현에도 활용된다. [그림 4]와 같이 각 프레임의 그림자를 2차원 이미지로 생성하고 이를 최종 렌더링 시 그래픽 가속기에 등록하여 적용하기 때문에 소프트웨어로 처리할 때보다 빠르고 자연스러운 그림자를 표현한다. 이외에도 환경에 대한 이미지를 미리 생성하고 이를 물체에 적용시키는 환경맵 기술이 있다.



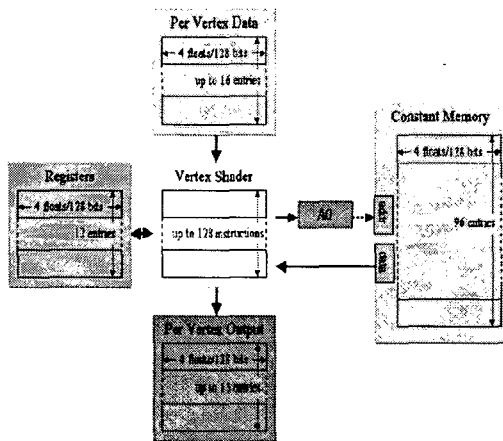
▶▶ 그림 3. 3개의 라이트맵과 적용 결과 영상



▶▶ 그림 4. 그림자 맵과 적용 예,
www.nvidia.com의
개발자 자료에서

물체의 질감과 광원과의 상호작용을 표현하기 위하여 셰이딩(Shading)은 중요한 컴퓨터 그래픽 기술이다. 종전의 셰이딩 방식은 그래픽 가속기에서 제공하는 몇 개의 셰이딩 방법 중의 하나를 선택하여 사용하거나 별도의 셰이딩 알고리즘을 소프트웨어로 구현하여 표현하는 방법이 있었다. 컴퓨터 그래픽 기술의 발전에 따라 그래픽 가속기를 프로그래밍하는 프로그램 가능한 셰이딩(Programmable Shading) 기술이 발전하여 사

용자는 게임 플레이에 필요한 특수효과 등을 표현하기 위하여 특별한 셰이딩 기법을 구현할 수 있다. DirectX 8.0부터 도입된 Programmable Shader는 Vertex Shader와 Pixel Shader로 나뉘어지는데, Vertex Shader는 그림 5와 같이 구성되어 있고 총 17 종의 연산을 가지며 한 번에 128개의 연산이 하나의 Vertex Shader에 등록되어 처리될 수 있다[3].



▶▶ 그림 5. Vertex Shader의 구성도, www.nvidia.com의 개발자 자료에서

2.2 애니메이션 기술

애니메이션 기술은 애니메이션이 적용되는 대상에 따라 분류하면 캐릭터 애니메이션과 객체 애니메이션으로 분류된다. 캐릭터 애니메이션은, 게임에서 사용되는 PC(Player Character), NPC(Non-Player Character), 몬스터에 적용되는 애니메이션으로, 일반적으로 캐릭터의 걷기, 뛰기, 공격 동작을 표현한다. 이와 달리 객체 애니메이션은 게임에서 사용되는 깃발, 폭포, 흔들리는 다리 등에 적용되는 애니메이션으로, 깃발의 흔들리는 현상과 폭포의 물 떨어지는 현상, 다리

의 흔들림을 표현한다. 캐릭터의 얼굴에 적용되어 얼굴 표현에 사용되는 애니메이션은 얼굴 애니메이션으로 구분된다. 또한 애니메이션 기술은 애니메이션의 동작 방식에 따라 Rigid Body 애니메이션, 관절체 애니메이션, 스킨 메쉬 애니메이션으로 구분된다. Rigid Body 애니메이션은 게임 객체를 구성하는 메쉬(Mesh)를 이동시킬 때, 각 정점(Vertex)에 이동 변환이 똑 같이 적용되는 경우이고 관절체 애니메이션은 이러한 메쉬의 이동이 캐릭터 관절의 시간에 따른 상대적인 위치에 따라 결정이 되는 방식이다. 하지만 이 두 가지 방식은 캐릭터의 움직임에 따라 캐릭터의 걸모양이 자연스럽지 못한 단점이 있다. 반면에, 스킨 애니메이션은 캐릭터의 움직임에 따라 각 관절체의 움직임이 캐릭터의 걸모양에 반영되어 자연스러운 움직임을 표현할 수 있다. 구체적으로 캐릭터 메쉬를 구성하는 각 정점에 대해 각 관절체가 갖는 가중치를 조절함으로써 가능하다.

스킨 메쉬 애니메이션은 적용 대상인 메쉬의 구성에 따라 단일 메쉬 스킨 애니메이션과 멀티 메쉬 스킨 애니메이션으로 구분된다. 단일 메쉬 스킨 애니메이션은 적용되는 메쉬가 하나로 이루어진 경우이고 다중 메쉬 스킨 애니메이션은 적용되는 메쉬가 여러 개로 이루어진 경우이다. 다중 메쉬 스킨 애니메이션을 사용하면, 일반적으로 스킨 애니메이션이 적용되는 갑옷이나 머리 등을 교체하는 게임플레이가 가능하다. 참고로 칼이나 방패는 스킨 애니메이션이 적용되지 않기 때문에 다중 메쉬 스킨 애니메이션이 적용되지 않는다. [그림 6]은 캐릭터의 바지를 교체한 예이다.



▶▶ 그림 6. 다중 매쉬 스킨 애니메이션을 이용한 캐릭터 바지의 교체

2.3 사운드 기술

소리의 전달 경로는 소리가 발생하는 공간에 의한 반사, 회절, 산란 등의 현상을 발생시키는 공간전달계와 인간의 머리와 귓바퀴에 의한 반사, 회절, 공진 등의 현상을 유발하는 머리전달계로 구분된다. 귀로 전달된 소리의 공간적 단서를 지각하는 주된 요인은 양귀에 도달하는 소리의 시간차와 소리의 세기차에 기인한다. 저주파의 경우 시간차로, 고주파의 경우 세기차로 지각하는 이중시스템으로 수행된다. 이외에 소리의 공간적 단서를 유발하는 요인들로는 시각적 효과, 머리 움직임 및 소리의 종류에 따른 친숙도 등을 들 수 있다.

음원(Sound Source)이 발생한 공간 내에 있는 청취자의 양쪽 귀에 마이크로폰을 각각 설치하여 녹음한 신호를 바이노럴 신호라 하며, 이 신호를 헤드폰으로 재생할 경우, 현장에서 직접 듣는 것과 같은 음상(Sound Image)을 지각할 수 있다. 여기서 음원이란 실제 물리적으로 음을 발생하는 객체나 위치를 말하며, 음상은 인간이 지각하는 감각상의 음원을 말한다. 음원과 음상은 공간적 특성이 반드시 일치하지 않으며, 음원과 음상이

일치할수록 좋은 음질의 입체음향이 구현되었다고 할 수 있다. 바이노럴 신호에는 음원의 위치, 방향뿐만 아니라 음원을 둘러싸고 있는 공간, 즉 음장(Sound Field)과 관련한 공간적 단서들이 포함되어 있다.

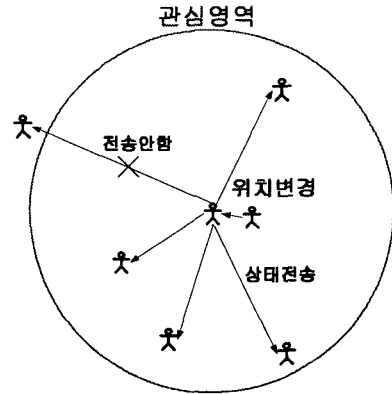
입체음향 기술은 사용한 채널 수에 따라 2채널 입체음향과 멀티채널 입체음향으로 구분된다. 2채널에 의한 방식은 인간이 두 귀로 음향을 지각하는 특성을 이용하여, 음상정위(Sound Image Localization)와 음장제어(Sound Field Control)에 의해 생성된 입체음향을 2채널에 의해서 재생하는 방식을 말한다.

이와 달리, 멀티채널에 의한 방식은 두 개 이상의 마이크로폰을 사용하여 음을 녹음하고, 다수의 스피커를 이용하여 음향의 공간적 분포를 재생하는 방식으로, 소리의 현장감을 향상시키기 위한 목적으로 많이 사용된다. 연주회장에서의 음향이 폭넓은 공간감을 주는 것은 반향음(벽면 또는 천장에서 반사되는 음)의 결과이다. 이 방식은 두 개의 스피커만으로 스테레오 음향을 재생할 경우, 스피커 두개가 청취자 정면에 있기 때문에 충분한 공간감을 느끼지 못하는 단점을 극복할 수 있다. 기본적으로 청취자의 전후 좌우에 2개 이상의 스피커를 배치(녹음의 경우도 여러 개의 마이크로폰을 배치)하여 음향이 청취자를 둘러싸는 서라운드 타입으로 재생한다. 멀티채널 기반의 입체 음향 기술은 AC-3, MPEG-2 오디오, DTS 등이 있다.

Ⅲ. 게임 서버 기술

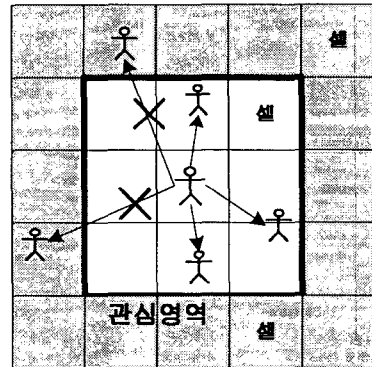
다른 장르의 게임과 달리 온라인 게임에서 게임 서버는 게임의 전체적인 진행을 총괄한다는

측면에서 아주 중요한 부분이다. 온라인 게임인 경우 한 서버 당 일반적으로 수천 명의 사용자가 이용하는 클라이언트의 통신을 담당하고 각 클라이언트에서 오는 게임 메시지를 처리하는 역할을 수행한다. 이런 관점에서 게임서버는 수많은 게임 메시지를 네트워크 딜레이를 최소화하며 실시간에 처리해야 한다. 아울러서 게임이 장기간 연속적으로 진행되기 때문에 게임서버의 안정성이 확보되어야 한다. 이러한 게임 메시지의 실시간 처리와 게임서버의 안정성을 위하여 관심 영역 기술과 이음새 없는 다중 게임 서버 기술이 시도되고 있다



▶▶ 그림 7. 거리 기반의 관심영역 기술

관심 영역 기술(AOI: Area of Interest)은 각 클라이언트에 보내지는 메시지의 양의 최소화하기 위한 기법으로 각 클라이언트의 주위에 있는 게임 객체의 상태와 그 변화만을 전송한다. 게임 객체의 주위에 있는 지 판단하는 방법에 따라 거리 기반 관심 영역 기술과 셀(Cell) 기반 관심 영역 기술로 나뉘어진다. 거리 기반 관심 영역 기술은 [그림 7]과 같이 클라이언트의 주 캐릭터와 각 객체간의 거리를 계산하기 때문에 계산 량이 증가한다는 단점이 있다. 이는 서버에서 처리되는 계산이기 때문에 서버의 부담이 되어서 바람직하지 않다. 이와 달리 셀 기반 관심 영역 기술은 [그림 8]과 같이 게임 공간을 셀로 구분하고 각 클라이언트의 주 캐릭터의 주위에 있는 객체의 상태와 그 변화만을 전송한다. 서버에서 객체의 정보를 셀 별로 관리해야하는 부담이 있어 메모리의 사용이 증가하지만 CPU 연산을 줄인다는 장점이 있다.

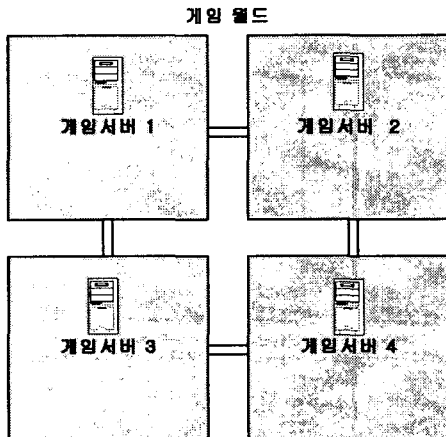


▶▶ 그림 8. 셀 기반의 관심영역 기술

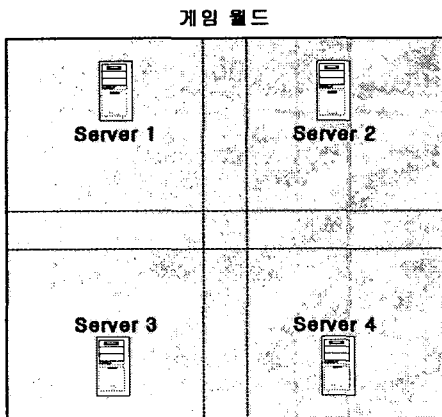
온라인 게임은 하나의 게임월드를 처리하기 위하여 다수의 게임 서버를 이용한다. 일반적으로 하나의 게임 서버가 물리적으로 처리할 수 있는 최대 클라이언트 수는 제한되어 있다. 이러한 제한 사항을 극복하기 위하여 게임월드를 다수의 작은 게임월드로 구성하고 각 게임월드를 독립적으로 동작하는 각각의 서버가 관리한다. 이 경우 각 서버당 게임월드를 서로 독립적으로 구성할 수 있고 아니면 작은 게임월드를 특정한 게이트웨이를 통해서 이동할 수 있도록 구성하기도 한다. 이와 반면에 이음새 없는 다중 분산 게임 서

버는 게임월드를 다수의 작은 게임월드로 나누어서 각 서버가 담당하며 서버가 담당하는 게임월드 간의 이동이 자유스럽다. 이는 그림4와 같이 각 경계영역에서 일정 영역을 이웃하는 서버가 동시에 관리함으로써 가능하다.

이외에도 게임 메시지의 양을 줄이기 위하여 객체의 상태가 변화하는 경우에만 게임 메시지를 처리하고 전송하는 기술인 데드 레커닝(Dead Reckoning) 기술이 있다.



▶▶ 그림 9. 포탈로 연결된 영역기반 분산 서버



▶▶ 그림 10. 이음새 없는 영역기반의 분산 서버

IV. 게임 인공지능 기술

게임 인공지능이란 고전적인 의미에서는 게임 내에서 컴퓨터에 의해 제어되는 캐릭터나 에이전트로 정의하기도 하지만 최근에는 좀더 구체적으로 스스로 생각할 수 있고 주변 환경이나 과거의 경험 등에 따라서 지능적으로 행동할 수 있는 자율성을 가진 캐릭터나 에이전트라고 정의하기도 한다. 그러면 인공지능이 게임에서 담당할 수 있는 역할은 무엇일까? 우선, 게임에서의 인공지능은 등장 캐릭터의 지능적인 행동을 구현함으로써 게이머가 조작하지 않는 NPC(Non-Player Character)들의 움직임을 자연스럽게 제어하거나 게이머의 상대 역할 또는 보조자 역할을 한다. 이 경우 게임의 핵심은 인공지능과의 대결이다. 인공지능은 게이머에게 무조건 이기는 것이 목적이 아니라 유사한 수준의 상대 역할을 수행해줄 수 있어야 한다. 또한 RPG와 같은 경우에 주인공의 보조자 역할을 해주거나 게임의 초보자를 이끌어 주는 역할을 하는 캐릭터가 될 수도 있다. 그리고 인공지능은 게임에서의 애니메이션 동작 제어를 담당하거나, 캐릭터가 현재의 위치에서 목적지까지 갈 수 있도록 이동경로를 찾아주는 역할을 하기도 한다.

게임에서의 인공지능은 한두 가지의 알고리즘만으로 구현하기는 힘들다. 게임의 수준과 장르에 따라서 다르겠지만 여러 가지의 인공지능 기법을 조합해 하나의 게임에 필요한 인공지능을 적용할 수 있는 것이다.

4.1 FSM

FSM(Finite State Machine)은 현재 가장 널리 사용되는 인공지능 처리 방식 가운데 하나이다. FSM이란 유한한 개수의 상태(state)를 이용

하여 NPC의 행동 양식을 표현하거나 게임 세계를 관리하는 방법이다. 상태란 행동처리를 위한 기본단위가 되며, 각 상태는 주어지는 조건에 따라서 다른 상태로 전이될 수가 있다. NPC의 행동 양식은 여러 개의 상태로 나누어지며 현재의 상태와 조건에 따라서 외부에 대처하는 방법이 결정된다. FSM은 이해하기 쉽고 구현도 어렵지 않아 특별히 뛰어난 인공지능을 필요로 하지 않는 대부분의 게임에서 사용된다. 그러나 복잡한 게임의 경우 상태의 수가 많아지게 되고 그에 따라 상태 다이어그램을 정리하게 어려워지며, 상태 변화를 가능하게 하는 외부 입력(조건) 루틴이 급속도로 복잡해진다. 또한 FSM을 게임의 상대방이나 NPC로 적용한 경우 게임 진행 중 유사한 경험을 몇 번 겪고 나면 인공지능의 행동 패턴을 예측할 수 있게 되므로 게임의 재미가 반감될 수도 있다. 이러한 문제점들을 해결하기 위하여 상태의 수가 많아질 경우 상태들을 몇 개의 그룹으로 묶어 계층적으로 구성하는 계층적 FSM를 사용하기도 한다. 이러한 계층적 FSM은 하프라이프(Half Life)라는 게임에서 사용되었다. 그리고 퍼지 FSM은 FSM에 퍼지(fuzzy) 이론을 접목하여 상태의 입력과 출력에 퍼지 함수를 적용하여 동일한 외부 상황에도 다른 출력을 얻을 수 있도록 함으로서 상대방이나 NPC의 행동을 예측하기가 어렵게 되어 보다 현실적인 게임을 즐길 수 있도록 해준다.

4.2 길찾기

게임에서 가장 자주 등장하는 현실적인 문제 가운데 하나는 현재의 위치에서 목적지 또는 목표물까지 가는 경로를 찾는 것이다. 예를 들어 전략 게임에서 목적지를 마우스로 알려주면 가장 빠른 지름길을 찾아 해당하는 장소로 이동해야

하는 문제로 전략 게임분만 아니라 거의 모든 게임에서 등장하는 문제이다. 이를 위하여 가장 널리 사용되는 방법은 에이스타(A*) 알고리즘을 이용하는 방법이다. 이 방법은 예상비용(Estimated cost)을 이용하여 경로에 대한 탐색 범위를 효율적으로 제한하는 방식이다. 맵(map)의 특성에 따라 다양하게 휴리스틱(heuristic)가 중치를 적용시킬 수 있는 장점으로 인하여 길찾기에서 다양하게 응용되고 있으나, 목표물까지 가는 도중에 길이 차단되었거나 폭파되어 끊겼을 경우와 같이 지형이 일시적으로 바뀌었을 때는 에이스타 알고리즘만으로 모든 길찾기 문제를 해결할 수 없으며 추가적인 방법이 요구된다. 길찾기 문제는 인공지능 문제 중에서 비교적 많은 연구가 진행된 분야이며 대부분 에이스타 알고리즘과 다른 방법을 함께 이용하여 해결한다.

방대한 지형에서의 길찾기일 경우에는 출발지점과 도착지점 사이의 거리가 멀어질 수 있다. 이런 경우에는 탐색해야 할 공간이 폭발적으로 증가하게 되므로 메모리의 낭비 등으로 탐색효율이 떨어지게 된다. 따라서 출발지점과 도착지점 사이의 중간 경유지를 생성하여 길찾기를 단계적으로 수행하는 계층적 길찾기 방법을 이용하여 해결할 수도 있다.

4.3 플로킹

플로킹(Flocking)이란 새나 벌, 어류 등과 같은 수많은 개체들이 무리를 지어 집단적으로 움직이는 모습을 흉내 내어 묘사하는 방법을 일컫는다. 플로킹의 대상이 되는 새나 벌, 물고기 등과 같은 개체를 보이드(boid)라고 부른다.

4.4 팀 인공지능

최근 게임은 온라인 기능이 강조되어 그 결과

로 참여하는 인원간의 팀워크를 중요시하게 되었다. 이에 기반하여 인공지능 기술에도 팀 인공지능이 중요한 이슈로 등장하였다. 1명의 지휘자와 다수의 팀원으로 이루어진 집단의 인공지능을 처리하는 방법으로 팀 배치, 지형정보를 이용한 전략적 이동, 개개인의 역할 분배 등이 중요한 요소로 이를 위하여 3가지 레벨로 나누어 처리할 수 있다. 우선 전략 레벨에서는 팀이 달성해야 할 목표에 의하여 통제되고, 팀 레벨에서는 각 목표를 달성하기 위한 계획에 의하여 통제되며, 개인 레벨에서는 각 개개인의 행동 규칙에 의하여 통제된다.

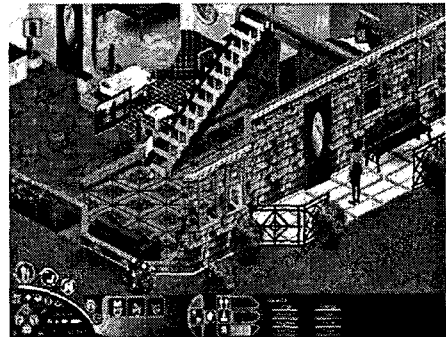
4.5 LOD AI

LOD AI(Level Of Detail AI)란 현재 스크린에 보이는 캐릭터의 인공지능은 구체적인 알고리즘을 적용하며, 보이지 않는 캐릭터의 인공지능은 보다 단순한 알고리즘을 사용하는 것을 말한다.

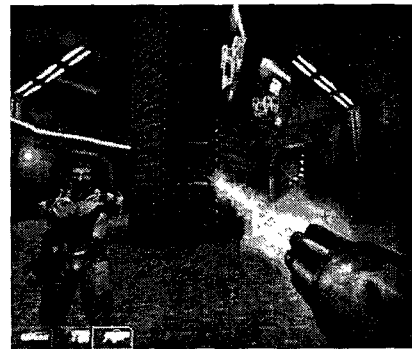
4.6 인공지능 기법을 사용한 게임

게임에서의 인공지능을 구현하기 위해서는 한 두 가지 방법만을 사용하기 보다는 게임의 수준이나 장르에 따라 몇 가지 방법을 조합하여 구현하는 것이 일반적이다. 블랙앤화이트(Black & White)에서는 인공지능이라는 기술이 적용되었는데, 캐릭터 자신의 욕구와 상태에 따라 사용자의 행동에 대한 반응이ダイナミック하게 나타날 수 있도록 하였다. 또한 사용자가 캐릭터에게 어떤 행동을 반복해서 보여주게 되면 그 행동을 똑같이 따라 하고 상이나 벌을 제시함으로써 선과 악을 구분하고, 교육이 가능하도록 하였다. EA사의 심시티 시리즈로 유명한 심즈(The Sims)는 윌라이트(Will Write)가 개발한 것으로 일상적인 실생활을 게임으로 옮겨놓은 것으로, 각 캐릭터

의 욕구가 있어서 그것들을 잘 충족시켜줘야 순조롭게 플레이를 할 수 있다. 심즈에서의 가장 특징적인 인공지능 요소는 캐릭터의 욕구에 기반한 인공지능 기법이 사용된 것과 어떤 행동에 대한 정보를 각각의 오브젝트가 갖고 있다는 것이다. 또한 인공지능 기법과 퍼지 기법을 조합하여 적용한 게임으로도 알려져 있다. 하프라이프에서는 몬스터나 적들의 일련의 행동을 스케줄로 묶어 처리함으로써 플레이를 할 때면 정말 만만한 상대가 아니다 라는 느낌을 받을 수 있도록 처리하였다. 또한 플로킹 방법을 적용하여 무리의 움직임이 자연스럽게 구현한 게임이기도 하다.



▶▶ 그림 11. 인공지능 기법이 적용된 심즈



▶▶ 그림 12. 인공지능 기능이 향상된 언리얼토너먼트

V. 결론

초고속 인터넷의 발전에 따라 온라인 게임은 게임 산업에서 중요한 위상을 차지하게 되었다. 온라인 게임 산업의 발전과 더불어 발전된, 온라인 게임의 네트워크 및 서버 기술은 향후 멀티플랫폼 게임 기술의 기초가 되어 게임 플랫폼이 통합되는데 기술적 기여를 하게 될 것이다.

그 동안 게임 산업은 게임 플랫폼에 따라 분리된 산업으로 성장하였고 그 기술적 영역도 확연히 구분되었다. 게임콘솔의 그래픽과 제작 기술은 PC 기반의 게임 그래픽 기술과 전혀 다른 형태를 취하고 있어서 PC 기반의 게임 기술을 갖고 있는 개발업체가 콘솔게임기를 위한 게임을 개발하기에 기술적 장벽이 높은 편이었다. 하지만, 게임플랫폼 별로 다소 차이가 있지만 한 게임당 수십억 원이 들어가는 게임 제작 예산의 급격한 증가와 한 종류의 게임플랫폼에서 상업적으로 성공한 게임 타이틀에 다른 게임플랫폼에서도 성공하는 경우가 많은 시장적 요인은 게임업체로 멀티플랫폼 게임 기술에 대한 수요를 증가시키고 있다.

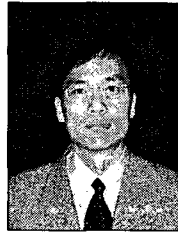
멀티플랫폼 게임 기술과 결합된 온라인 게임 기술은 멀티플랫폼을 동시에 지원하는 게임의 개발 및 서비스를 가능하게 될 것이다. 이러한 기술을 이용한다면 수년 내에 PC와 Playstation3, Xbox360, PSP, GameCube, 휴대폰 등에서 동시에 사용이 가능한 게임이 출현할 것이고, 이는 언제(Any Time), 어디서나(Any Where), 어떤 단말기(Any Device)로 플레이 가능한 3-A Digital Entertainment를 실현하게 될 것이다.

참고문헌

- [1] 한국전자통신연구원, "온라인 3D 게임 엔진의 표준화 연구에 대한 보고서", 2001년 12월
- [2] 정보통신교육원, "게임 엔진 개발", 2002년 12월
- [3] "nVidia 개발자 사이트", <http://developer.nvidia.com>
- [4] "DirectX API Reference", <http://www.microsoft.com>

저자 소개

● 임충규(Choong-Gyu Lim)



- 1988년 : 서울대학교 수학교육학과 (이학사)
- 1990년 : 서울대학교 대학원 수학교육학과(교육학석사)
- 1990년~1992년 : 현대전자산업주식회사 근무
- 1993년~1998년 : 미국 루이지애나주

립대학교 대학원 전산과학과(Ph.D)

- 1999년~현재 한국전자통신연구원(선임연구원)

<관심분야> : 컴퓨터 그래픽스, 컴퓨터 게임, 기하 모델링, Computer Aided Geometric Design

● 이현주(Hun-Joo Lee)



- 1991년 : 중앙대학교 컴퓨터공학과 (공학사)
- 1993년 : 중앙대학교 컴퓨터공학과 (공학석사)
- 1998년 : 중앙대학교 컴퓨터공학과 (공학박사)
- 2001년~2002년 : Iowa State

University(Post-Doc.)

- 1998년~현재 : 한국전자통신연구원(선임연구원)

<관심분야> : 인공지능, 게임엔진, 가상현실