

군중 장면 연출을 위한 마야 플러그인 도구 개발

이상곤, 남양희
세종대학교 컴퓨터공학과
e-mail : sangkon78@hanmail.net, yhnam@sejong.ac.kr

Development of Maya Plug-In for production of Crowd Scene

Sangkon Lee, Yanghee Nam
Dept. of Computer Science, Se-Jong University

요 약

오늘날 영화, 게임, 애니메이션 등 다양한 분야에서 사용되고 있는 군중 장면은 모델러의 많은 수작업을 필요로 한다. 모델러에게 있어 배경과 배경 물체들 그리고 수많은 에이전트들을 적절히 배치해야 하며, 매 프레임마다 이들 간에 충돌이 없도록 적절히 움직여 주어야 하는 수작업은 비능률적이다. 따라서 본 논문에서는 군중 장면 연출 도구를 제작하여 군중들을 배치할 수 있는 방법과, 배경 물체들과 군중 사이 그리고 군중을 이루는 에이전트들 사이의 충돌 회피 방법을 제시한다. 이를 통해 모델러는 군중을 일일이 배치하고, 매 프레임마다 군중을 적절히 움직여 주어야 하는 수작업에 드는 시간을 모델링에 투자함으로써 작업 효율을 높일 수 있다. 또한 본 논문의 군중 장면 연출 도구는 마야 플러그인으로 개발되어 대표적인 모델링 도구인 마야와 연동하여 군중 장면을 연출할 수 있는 장점을 가진다.

1. 서론

오늘날 군중 장면은 영화, 게임, 애니메이션 등 다양한 분야에서 사용되고 있다. 소수의 캐릭터들을 통해 결과를 얻는 정밀한 애니메이션과 달리, 군중 장면은 군중들의 배치 및 군중들 사이의 다양한 상호작용과 충돌 회피 문제가 중요하게 부각된다.

군중 장면의 예로 많은 사람이 오가는 도심가를 모델링 한다고 할 경우 모델러는 먼저 도심가의 지형과 주요 건물들을 모델링 해야 한다. 그 후 원하는 위치에 군중들을 배치시키고, 매 프레임당 많은 군중들을 일일이 애니메이션 시켜야 한다. 이러한 방식은 군중들의 수가 많아지면 많아질수록 그리고 배경이 복잡하면 복잡할수록 더욱 많은 시간을 요구하게 된다. 그렇기 때문에 빠른 시간에 결과를 얻어야 하는 영화, 게임, 애니메이션 등의 산업전선에서는 커다란 문제가 될 수 있다. 따라서 작업의 효율과 생산성 향상을 위해서는 모델러에게 쉽게 군중 장면을 연출할 수 있도록 제공해 줄 수 있는 군중 장면 연출 도구

가 필요하다.

군중 장면 연출 도구가 군중 장면 연출을 제공하기 위해서는 군중 장면에 필요한 기본 제어 요소를 만족해야만 한다.

- 1) 에이전트의 기본 정보 : 에이전트는 군중을 이루는 최소단위로서 기본적으로 위치 정보, 행동 정보, 자신이 속한 그룹 정보 등을 가져야 한다.
- 2) 군중의 배치 : 군중 장면에 필요한 많은 에이전트들을 모델러가 일일이 배치한다는 일은 비효율적이기 때문에 군중을 배치하는 방법이 필요하다. 군중 장면에서 군중은 연관성 있는 에이전트들이 모여 있는 것이라는 특성을 가진다. 따라서 모델러가 군중을 배치할 때에 연관성 있는 에이전트들을 원하는 수만큼 한데 묶어 그룹 단위로 생성하고 배치할 수 있다면 더욱 효과적으로 군중 장면을 연출할 수 있다.
- 3) 군중의 목표 : 군중에 속한 에이전트들은 공동된

목표를 향해 움직여야 한다. 따라서 에이전트들이 공동으로 참조할 수 있는 목표들을 지정할 수 있어야 한다.

- 4) 자동적 충돌 감지 및 충돌 회피 : 군중 장면을 플레이백 할 때에 에이전트와 배경 물체간 그리고 에이전트와 에이전트간의 충돌이 필연적으로 발생하게 된다. 군중 장면 연출 도구는 모델러의 수작업이 아닌 알고리즘에 의한 자동적 충돌 회피를 제공함으로써 충돌이 없는 군중 장면 연출을 보장할 수 있어야 한다.

본 논문은 위의 제어 요소를 고려하여 군중 장면 연출 도구(Crowd Scene Production Tool : CSPT)를 개발하였다. CSPT는 군중의 배치 문제를 해결하기 위해 에이전트의 수와 배치할 위치 그리고 배치할 반경을 입력으로하여 가우시안 분포 형태로 군중을 생성해주는 가우시안 군중 배치 방법을 사용한다. 이때 배치된 각 에이전트들은 같은 연관성을 가지므로 하나의 군중으로 취급되며, 이 군중은 모델러에 의해 입력된 이동속도, 목표 위치들 등의 값으로 초기화된다. 충돌 감지 및 회피를 위해서 CSPT는 충돌을 정면 충돌과 측면 충돌의 두 가지 경우로 분류하여 회피하는 방법을 사용한다. 또한 CSPT는 마야 플러그인으로 개발되기 때문에 대표적인 모델링 도구인 마야와 연동하여 군중 장면을 연출할 수 있는 장점을 가진다.

본 논문은 2장에서 기존의 연구를 살펴본다. 3장에서는 본 논문에서 제시하는 군중 장면 연출을 제공하기 위한 CSPT의 시스템 구조를 설명한다. 4장에서는 군중의 배치 및 군중 정보를 갱신하는 방법에 대해 알아본다. 5장에서는 배경 물체와 에이전트간, 그리고 에이전트들 상호간에 발생하는 충돌을 회피하는 알고리즘에 대해 설명하고, 6장에서 실험 결과 및 결론을 제시한다.

2. 기존 연구

사람과 동물의 행동을 분석하여 재현해 보려는 시도가 로보틱스, 인공지능, 인공생명 등 다양한 분야에서 있었다. 기존의 행동에 대한 연구는 개인행동 연구와 군중행동 연구로 나눌 수 있다.

개인행동 연구는 군중이 아닌 소수의 에이전트를 대상으로 한다. 에이전트는 독립적으로 존재하며, 주위의 상황을 인식하여 자신의 행동선택 방법에 따라 판단하여 행동을 하게 된다. 대표적으로 Thalmann은 각 에이전트마다 synthetic vision 개념을 통한 주변 인식 기능을 부여하여 인식된 주위의 장애물 또는 다른 에이전트들과의 충돌을 회피하도록 행동하는 시스템을 제안하였다[4].

군중행동 연구는 사람과 동물 등 다방면에 걸쳐 연구되었다. 집단적인 행동을 재현하기 위해 에이전트는 군중에 종속적으로 존재하며, 자신이 속한 군중에 대해 소속감을 가진다. 소속감이 강할 수록 개별적인 행동보다 군중적인 행동을 보이게 된다. 대표적으로 Reynolds는 BOID 시스템을 통하여 새의 무리행동을

표현하였다[1][2]. Reynolds는 위치, 이동 방향, 이동 속도의 정보를 가지는 그룹 내에 위치, 이동 방향, 이동 속도, 그룹 인덱스와 소속감 정보를 가지는 새들을 종속시켰다. 그룹에 속한 각각의 새들은 소속감의 정도에 따라 군중적인 행동과 개별적인 행동들을 보이게 된다. Musse는 사람 군중 행동 모델을 연구하였다[3]. 그는 군중에 속한 사람 중 가장 소속감이 큰 사람을 그 군중의 리더로 정하였다. 리더를 제외한 군중에 속한 다른 사람들은 리더를 따라가게 되며, 리더는 소속감의 정도에 따라 바뀔 수 있도록 함으로서 군중 행동을 연출하였다.

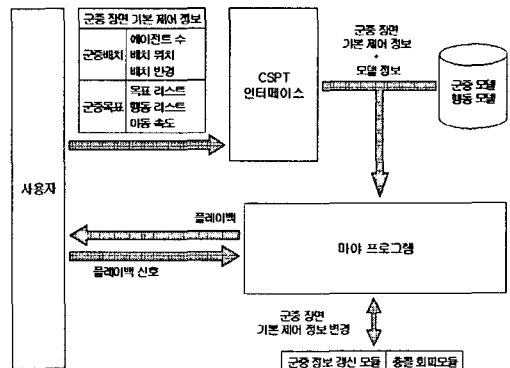
현재 상업적으로 군중 장면 연출을 위해 널리 쓰이고 있는 3DSMAX 플러그인 캐릭터 스튜디오는 군중 장면 연출을 위하여 다양한 기능을 제공하고 있다. 캐릭터 스튜디오는 개인적인 행동과 집단적인 행동을 모두 지원하고 있는데, Delegate라 불리는 군중의 최소단위를 위치, 이동방향, 이동속도, 가속도, 행동 리스트, 가중치 등의 다양한 정보들을 통해 조절할 수 있다. 또한 여러 개의 Delegate들을 한데 묶어 팀을 구성하여 Delegate와 동일한 정보 즉 방향, 속도, 가속도, 행동 리스트, 가중치 등을 적용해 군중 장면을 연출할 수 있다. 또한 군중을 원하는 위치에 원하는 수 만큼 흩뿌려 생성해주는 군중 배치 기능을 제공하여 빠르게 군중을 만들 수 있도록 제공한다.

본 논문의 CSPT는 군중행동 패턴 연구의 관점을 따라 개발되었다.

3. 시스템 구조

군중 장면 연출 도구는 군중 장면의 기본 제어 요소들을 모델러가 입력할 수 있도록 제공해야 한다. 기본 제어 요소인 에이전트의 기본정보 즉 현재 위치, 이동 방향, 이동 속도, 소속된 그룹, 행동 리스트를 입력할 수 있어야 하며, 군중의 배치 정보 즉 사람 수, 배치할 위치, 배치할 반경을 입력할 수 있어야 한다. 또한 내부적으로 매 프레임마다 각 에이전트와 군중 정보를 재계산하는 모듈과 충돌 회피를 제공하는 모듈이 있어야 한다.

본 논문에서 위의 요구조건을 고려하여 아래 그림 1과 같이 CSPT를 구성하였다.



[그림 1] 군중 장면 연출 도구의 시스템 구조

모델러는 CSPT 인터페이스를 통해 사람 수, 배치 위치, 배치 반경을 입력하므로 군중을 배치하게 된다. 이 때에 모델러가 입력한 군중 정보 즉 군중의 이동 속도, 목표 리스트, 행동 리스트를 통해 현재 생성된 군중을 초기화한다. 모델러는 이러한 작업을 반복하며 독립적인 군중들을 임의의 개수만큼 생성할 수 있다. 모든 작업을 마치고 플레이백 할 때에 군중정보 갱신모듈과 충돌회피 모듈이 동작하여 군중 장면 기본 제어 정보들을 변경시키므로 충돌없는 군중 장면의 연출을 가능하게 한다. 군중 배치와 군중 정보 갱신 및 충돌 회피 방법을 4, 5장에서 알아본다.

4. 군중 배치 및 군중 정보 갱신

4.1 가우시안 분포를 이용한 군중 배치

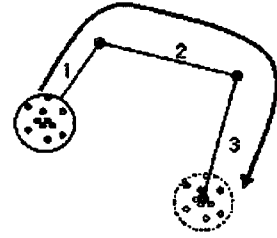
모델러가 수 많은 에이전트들을 일일이 배치시켜야 한 한다면 많은 시간과 노력이 필요하므로 비효율적이다. 따라서 군중을 배치시킬 수 있는 방법이 필요하다. 본 논문에서는 군중은 연관성 있는 에이전트들이 인접한 곳에 모여 있는 특성을 가짐을 이용하였다. 예를 들어 버스를 기다리는 사람들은 버스를 기다린다는 연관성을 가지게 되고, 이를 위해 버스 정류장 부근에 모여 있게 되는 것을 의미한다. 이러한 군중의 특성을 감안하여 한 곳을 중심으로 서로 인접하게 에이전트를 배치하기 위해 본 논문에서는 가우시안 분포를 이용하였다. 가우시안 분포를 이용한 군중 배치 방법은 모델러로 하여금 배치할 사람의 수와 배치할 위치 그리고 배치할 반경을 입력으로 받는다. 이 정보를 바탕으로 군중 배치 알고리즘은 정해진 반경내에서 가우시안 분포에 따라 군중을 배치하게 된다. 아래 그림 2 는 가우시안 분포를 이용한 군중의 배치를 보여준다.



[그림 2] 가우시안 분포를 이용한 군중의 배치

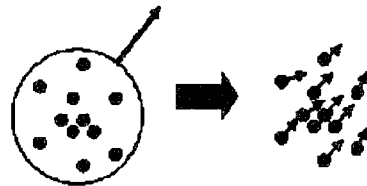
4.2 군중 정보 갱신

4.1 에서 모델러는 군중을 배치할 때에 배치할 위치와 군중이 이동하게 될 목표 위치들을 입력하게 된다. 이 정보들로부터 군중이 움직여야 할 전체적인 이동 경로를 알 수 있다. 아래 그림 3 은 군중의 이동 경로를 보여준다. 그림에서 군중은 1, 2, 3 의 경로를 따라 움직이게 된다. 이때 군중에 속한 에이전트들 각각의 정보들은 군중의 이동 경로를 바탕으로 결정된다. 즉 군중의 중심이 군중의 이동 경로를 따라 움직일 때에 매 프레임마다 군중의 이동 방향과 이동 속도가 변하게 된다.



[그림 3] 군중의 이동경로

아래 그림 4 는 군중의 정보에 따라 군중에 속한 에이전트들의 값이 갱신된 그림을 보여준다.

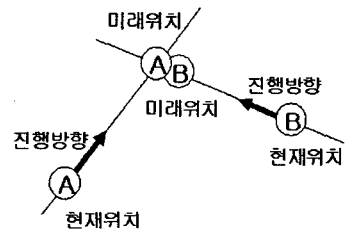


(1) 군중의 이동방향 (2) 에이전트의 이동방향
[그림 4] 에이전트 정보의 갱신

군중에 속한 각각의 에이전트들에게 군중의 정보를 통해 이동방향과 속도 정보를 할당할 때에 약간의 테크닉이 필요하다. 즉 모든 에이전트에게 동일한 군중의 정보를 할당하면 모두 같은 방향과 같은 속도로 움직이는 비정상적인 군중의 움직임을 보이게 된다. 따라서 방향과 속도의 값에 랜덤한 변화를 주어서 엇비슷한 값을 생성하여 각각의 에이전트에 할당함으로써 군중적인 행동을 취하지만 조금씩 다른 군중 장면을 연출할 수 있게 된다.

5. 충돌 회피

4.2 에서 값을 할당 받은 에이전트들은 각각의 이동 방향과 이동속도를 이용하여 매 프레임마다 이동하게 된다. 이때 필연적으로 충돌이 발생하게 된다. 본 논문에서는 Reynolds 의 충돌 회피 방법(Aligned Collision Avoidance) 에서 임의의 방향으로 진행하는 에이전트와 에이전트간에 충돌을 판별하고 회피하는 방법을 제시하였다.

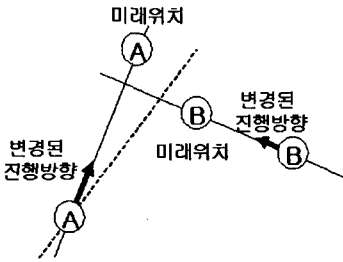


[그림 5] 비정렬 충돌 회피에서의 충돌 판별

에이전트 A 와 B 는 현재 프레임에서 다음 프레임으로 변화할 때에 어느 방향, 어느 정도 속도로 움직여

야 할지를 결정해야 한다. 이를 위해서는 미래에 자신이 존재할 위치와 미래에 다른 에이전트가 존재할 위치가 충돌을 하는지 검사를 함으로서 알 수 있다. 즉 모든 에이전트에 대해 각 에이전트의 현재 위치에서 진행방향으로 동일한 시간동안 움직였을 때에 충돌을 일으키는지 검사하는 것이다. 위의 그림 5는 에이전트 A와 B가 미래에 충돌을 일으키게 됨을 판별한 그림을 보여준다.

충돌을 판별한 후 충돌을 회피하는 방법은 의외로 간단하다. 그림 5를 예로 들어 설명하면, 에이전트 A의 관점에서 B와 충돌을 일으키지 않기 위해서는 현재의 진행방향을 좀더 왼쪽으로 수정하고, 이동속도를 더 빠르게 하면 된다. 반대로 B의 관점에서 A와 충돌을 일으키지 않기 위해서는 이동속도를 줄이면 된다. 이것을 그림 6에서 보여주고 있다. 에이전트 A의 진행방향은 왼쪽으로 에이전트 B의 진행방향은 짧아졌음을 볼 수 있다. 에이전트 A와 B의 미래 위치를 볼 때에 충돌이 일어나지 않게 된 것을 볼 수 있다.

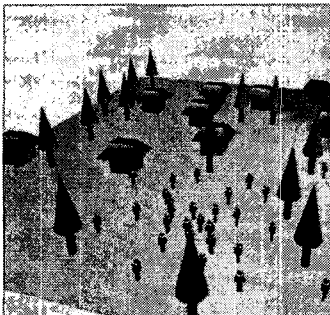


[그림 6] 에이전트 A와 B의 이동경로 수정

에이전트와 장애물간의 충돌도 마찬가지로 구할 수 있다. 에이전트의 미래 위치가 장애물의 바운딩 스피어와 충돌을 일으킬 경우를 판단하여 장애물은 움직이지 않으므로 에이전트의 방향만 변경해주면 충돌을 피할 수 있다.

6. 실험 및 결론

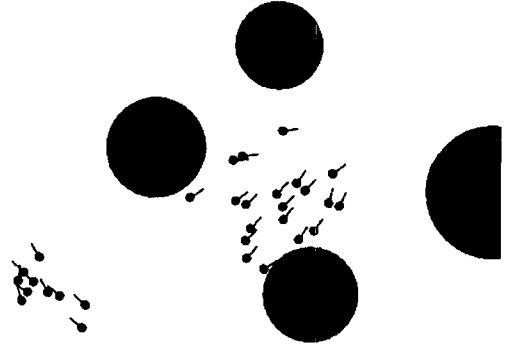
본 논문이 제시한 군중의 배치 방법과 충돌 회피 방법은 Maya 4.0 API와 Visual C++ 6.0을 사용하여 실험하였다[5][6].



[그림 8] 가우시안 군중 배치 결과 영상

위의 그림 8은 가우시안 군중 배치 방법을 사용하여 군중을 배치한 결과이다.

아래 그림 9는 여러 군중들이 목표 리스트를 따라 이동할 때에 발생하는 에이전트와 에이전트간 그리고 에이전트와 장애물간의 충돌 회피를 보여준다.



[그림 9] 충돌 회피 결과 영상

위의 그림 9에서 큰 이 장애물의 바운딩 스피어이고 작은 원이 에이전트의 바운딩 스피어이다. 2개의 군중이 각각 자신의 목표 리스트를 향해 가고 있는 결과 영상이다.

결론적으로, 본 논문에서는 군중 장면 연출을 위한 군중 장면 연출 도구인 CSPT를 제시하였다. CSPT를 구현하기 위해 군중을 이루는 에이전트를 쉽게 배치하게 하는 가우시안 분포방법을 제시하였다. 또한 군중 내의 에이전트의 정보를 갱신하는 군중 정보 갱신 방법과 Reynolds가 제시한 비정렬 충돌 회피를 이용한 충돌 회피 방법을 제시하였다. 향후 군중의 목표 리스트를 통해 얻어진 군중의 이동경로를 따라서 에이전트들을 움직이는 방법에서 더 나아가 더욱 다양한 행동 패턴을 군중 또는 에이전트 각각에 적용하여 군중장면을 연출할 수 있도록 할 계획이다.

* 서울시 중소기업청의 2001년 산.학.연 공동기술개발과제의 일환으로 (주)ZERON DIGITAL의 협력하에 진행됨

참고문헌

- [1] C.W. Reynolds "Flock, herds and schools : A distributes behavioral model", Computer Graphics, pp25-34, July 1987.
- [2] C.W. Reynolds "Steering Behaviors for Autonomous Characters" Game Developers Conference, pp763-782, 1999.
- [3] S. R. Musse and D. Thalmann "A Model of Human Crowd Behavior : Group Inter-Relationship and Collision Detection Analysis", SIGGRAPH 98, Course notes 80, 1998.
- [4] D. Thalmann "A Vision-Based Approach to Behavioral Animation", Journal of Visualization and Computer Animation, Vol.1, No1, 1990
- [5] Alias | Wavefront General User Association Site, <http://www.highend3d.com/>
- [6] Maya API Developer's Manual 4.0