

회피규칙과 MFC를 이용한 3D FPS 게임의 오브젝트 탐색 연구

최원태* · 최성호* · 김해용* · 박창민*

*영산대학교 게임·콘텐츠학과

A Study on the Object Search in 3D FPS Games Using Avoidance Rule and MFC

Won-Tae Choi* · Sung-Ho Choi* · Hae-Yong Kim* · Chang-Min Park*

*Dept of Game · Contents, Young San University

E-mail : nowwind@msn.com, sspk1@nate.com, darknox@naver.com, cmpark@ysu.ac.kr

요 약

MFC는 플레이어 오브젝트를 중심으로 영역을 만들어 상대 오브젝트의 유무를 판단한다. 그러나 게임에서 상관없는 오브젝트들까지 모두 탐색하여 게임의 속도를 저하시키고 위험성을 인식하는 단점이 발생한다. 본 논문에서는 이러한 단점을 보완하기 위하여 MFC를 이용할 때 플로킹의 회피규칙을 npc 오브젝트에게 적용 하였다. 따라서 npc 오브젝트는 시야의 개념을 가지게 되고 게임의 속도 저하를 막을 수 있는 방법을 제안하였다. 이때 제안된 2가지 조건을 만족해야 한다. 제안한 방법은 향후 3차원 FPS 게임의 발전에 주요한 역할을 할 것이다.

ABSTRACT

The MFC makes the region in the player object center and it judges a relation the object. But it searches the objects which are not relation to game. Consequently, it decreases the speed of game and it recognizes a dangerous characteristic. In this paper, in order to complement a weak point, it applied the avoidance rule of the flocking in npc object. Consequently, the npc object has the concept of range of vision and proposed the method which is possibility of closing the speed decrease of game. This time 2 condition which is proposed must be satisfactory. The method that proposed will play an important role in development of a 3D FPS games.

키워드

변형된 절두체 컬링(modified frustum culling), RPG(Role Playing Game), 플로킹(flocking), 회피규칙(avoidance rule), 오브젝트 탐색(object search)

1. 서 론

실시간 3차원 게임 월드에서 수많은 폴리곤들과 객체를 렌더링 하는 것은 게임의 속도에 많은 영향을 준다. 따라서 3차원 그래픽을 구현하는 단계에서 실제로 카메라의 시야 범위에 포함되는 폴리곤과 객체들만 렌더링 하는 기법으로 절두체 컬링(frustum culling)[1,5]을 이용한다.

그리고 변형된 절두체 컬링(Modified Frustum

Culling, MFC)[2]은 FPS (First Person Shooting) 게임에서 위협이 되는 오브젝트들을 미리 인식할 수 있게 하여 새로운 전략 시스템을 구사할 수 있는 방안을 제시 하였다. 이것은 플레이어의 관점에서 카메라 시야에 들어오지 않는 적 오브젝트들에 대한 위험성을 미리 감지할 수 있는 장점이 있다.

하지만 MFC을 다른 장르의 게임에 적용을 시켜 보았을 때 FPS장르에서 장점으로 사용되었던

오브젝트 탐색이 전체 게임 진행을 다소 지연시키는 문제점을 발생시켰다. 특히 액션 RPG(Action Role Playing Game)와 같은 장르에 MFC를 적용했을 때 공격 성향을 띄지 않고 지나가는 사소한 오브젝트들까지 탐색을 하여 게임의 속도감을 떨어뜨렸다.

한편, 플로킹(flocking)[3]의 조타행동(steering behavior)들 중에서 회피(Avoidance)기법은 게임에서 적 오브젝트가 바라보는 시야를 생성시켜 MFC를 이용한 오브젝트의 탐색에서 발생하는 문제점들을 줄일 수 있다.

본 논문에서는 이러한 플로킹의 회피기법을 게임에 MFC 적용 시 위험성을 단계적으로 나타내게 하여 가장 위험한 오브젝트만 탐색하고 공격 성향을 띄지 않는 오브젝트는 탐색하지 않음으로써 게임의 속도를 증가 시키는 방법을 제안한다.

II. 변형된 절두체 컬링과 플로킹

[2]에서 제안된 MFC는 플레이어 오브젝트가 일정 크기의 영역을 만들어 적 오브젝트들을 탐색하는 방법을 제시하고 있다.

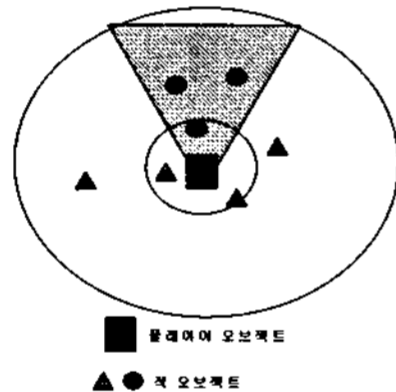


그림 1. MFC를 이용한 오브젝트 탐색

그림 1.에서 알 수 있듯이 기존의 절두체에서는 삼각형의 적 오브젝트들을 탐색하지 못하고 빗금 영역의 둥근 모양의 적 오브젝트들만 탐색을 한다. 그러나 MFC를 이용하면 빗금 영역에 있는 적 오브젝트들을 화면 렌더링 할 뿐만 아니라 플레이어 오브젝트 뒤에 위치한 적 오브젝트들의 위치를 계속적으로 탐색하여 위험도를 감지할 수 있게 하여 게임의 전략 및 전술을 세우는데 도움을 주며 게임의 긴박성을 한층 고조시키는 역할을 한다.

한편, 플로킹은 집단행동을 흉내 내는 기법으로 게임의 현실감 있는 환경을 제공해 주는 수단으로 널리 이용되고 있다. 이러한 집단행동을 조타행동이라고 하며 그림 2에서 나타내고 있는 바와 같이 4가지 규칙을 따른다.

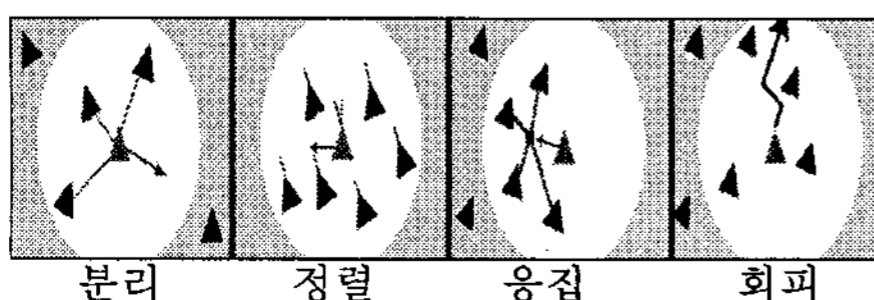


그림 1. 플로킹의 조타행동 규칙

이때 분리(Separation), 정렬(Alignment), 응집(Cohesion), 회피는 각각 다음과 같은 행동을 표현한다. 먼저 분리는 주변 오브젝트들과 충돌하지 않도록 방향을 돌리는 행동, 정렬은 주변 오브젝트들과 같은 방향을 가리키도록 하는 행동, 응집은 주변 오브젝트들과의 평균위치(질량중심)쪽으로 방향을 돌리는 행동 그리고 회피는 지역 장애물이나 적들에 부딪치지 않도록 반향을 조정하는 행동을 나타낸다.

III. MFC와 플로킹을 이용한 오브젝트 탐색

MFC는 주변의 적 오브젝트들을 계속적으로 체크함으로써 여러 전략적 요소를 만들 수 있지만 액션 장르의 게임에서는 오히려 속도감이 있는 플레이를 하지 못하게 하는 요소로 작용하는 단점이 있다. 따라서 플로킹의 조타행동에서 회피기법을 이용하여 변형된 절두체의 단점인 무조건적인 오브젝트 탐색으로 인한 게임의 긴박감을 저하시키는 문제점과 적 오브젝트들에 대한 위험도 체크를 통한 부담감을 줄일 수 있다.

3.1 플로킹을 이용한 탐색

플로킹의 조타행동 중에서 오브젝트가 이동을 할 때 회피기법에 의하여 오브젝트가 바라보는 방위에 다른 오브젝트가 있을 때 방향을 전환하여 충돌을 방지하는 행동이다. 이러한 회피 기법은 오브젝트의 방위에 따른 방향 전환을 하는 개념이지만 본 논문에서는 이러한 회피기법의 방위를 적 오브젝트가 바라보는 시야의 개념을 가지고 같은 방위의 시야에 플레이어 오브젝트가 있는지를 판단 하고자 한다.

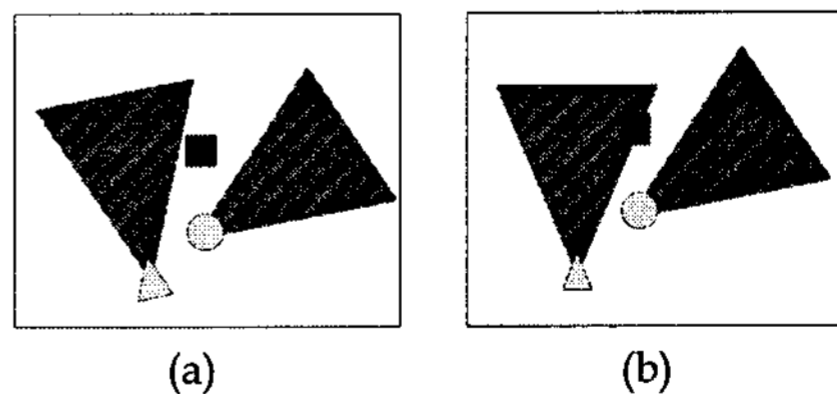


그림 3. 시야의 개념을 가진 회피행동에 따른 오브젝트 인식

그림 3.(a)은 삼각형(적) 오브젝트가 사각형(플레이어) 오브젝트를 인식하기전을 보여준다. 빗금 친 영역은 적 오브젝트의 시야 범위를 나타낸다. 이때 적 오브젝트의 시야에 플레이어 오브젝트를 들어오지 않았기 때문에 공격 성향이 나타나지 않고 원형(적) 오브젝트 역시 인식을 하지 않았기 때문에 아무런 변화가 없다.

이와는 반대로 그림 3.(b)는 적 오브젝트가 플레이어 오브젝트를 시야에 넣어 인식했을 때를

나타낸다. 이때 적 오브젝트는 시야안의 플레이어 오브젝트와의 거리를 단계적으로 체크하여 공격의 유무를 판단하게 된다. 반면 플레이어 오브젝트 역시 인식된 적 오브젝트에 의해 위협성을 감지하게 된다. 하지만 등근(적) 오브젝트는 아직 플레이어 오브젝트를 인식 하지 못하였기 때문에 공격 성향을 띄지 않으며 플레이어 오브젝트 또한 등근(적) 오브젝트에 대해선 위협도를 체크 하지 않는다.

3.2 MFC와 플로킹을 이용한 탐색

적 오브젝트의 시야 범위에서 일정거리 안에 들어오는 플레이어 오브젝트의 단계적 판정을 위해 MFC 방법을 적 오브젝트 좌표와 동일시 한다. 따라서 자신이 아닌 다른 오브젝트가 다가오게 되면 그 오브젝트의 위치 값이 자신과 어느 정도 떨어져 있는지를 [2]의 수식을 이용하여 단계별 위협성을 판단 한다.

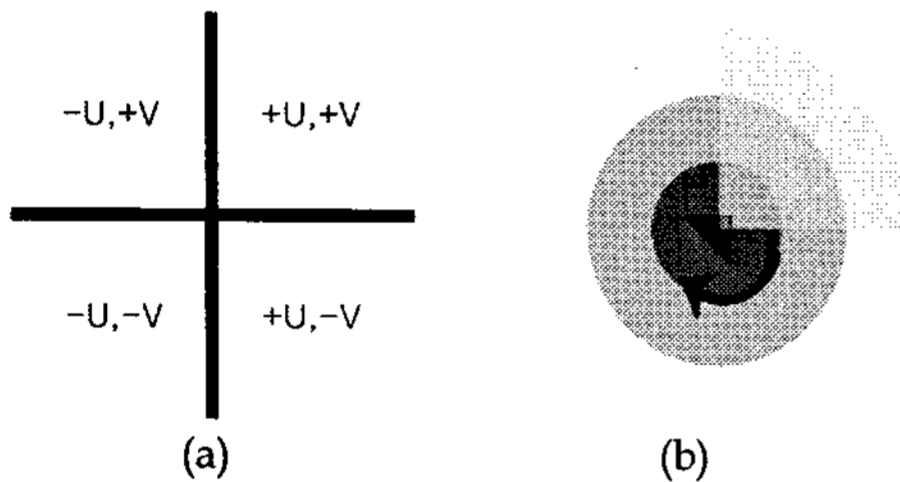


그림 4. 회피행동을 이용한 오브젝트 탐색

본 논문에서 이용한 플로킹 조타행동 규칙의 하나인 회피 기법으로 적 오브젝트를 탐색할 때 인식하는 범위를 4분면으로 나누었다. 그 이유는 오브젝트 탐색을 할 때 시야의 영역을 뚜렷하게 나타내기 위함이다. 먼저 그림4. (a)에서 U와 V는 3D상의 X, Y값을 정한 것이며 하나의 오브젝트는 일정 거리 안을 4분면으로 나누어 자신의 위치를 0으로 하고 1,2,3,4분면 중 1분면은 +U+V, 2분면은 +U-V, 3분면은 -U-V, 4분면은 -U+V로 설정을 한다. 플레이어 오브젝트는 이 4분면들을 차례로 체크를 하여 상대 오브젝트를 탐색하게 된다. 이때 거리에 따라 경고-1, 경고-2, 위험 등으로 단계를 구분하게 된다.

먼저 임의의 플레이어 오브젝트가 3분면에 있다고 가정하면 적 오브젝트는 1분면부터 차례로 검색이 들어가며 3분면을 검색하기 전까지는 플레이어 오브젝트를 인식하지 못한다(그림 4. (b)).

MFC 만을 적용한 플레이어 오브젝트는 자신의 주변을 모두 탐색하지만 회피기법이 적용된 적 오브젝트의 시야범위에 들어있지 않을 경우는 플레이어 오브젝트는 적 오브젝트를 탐색하지 않게 된다. 따라서 무조건적인 오브젝트 판정을 막기 위하여 조건을 부여하게 되면 상대 오브젝트와 자신 오브젝트가 상반 되는 위치에 놓여야 한다.

즉 상대 오브젝트가 위치한 4분면 범위와 자신 오브젝트가 위치한 4분면 범위가 반대로 되어야 한다는 첫 번째 조건이 생긴다.

한편, 그림5와 같은 경우에는 탐색을 하면 안 된다. 그림5.(a)은 사각형 오브젝트와 삼각형 오브젝트는 서로 상반되는 분면에 존재하지만 탐색과 판단의 거리에서 외부에 있기 때문에 서로 인식조차 불가능 하다. 이것은 두 번째 조건이 되며 서로 인식할 수 있는 거리에 있어야 된다는 것을 의미한다. 반대로 그림 5.(b)에서는 사각형 오브젝트에서 판정은 가능하지만 삼각형 오브젝트에서는 탐색범위가 상반되는 분면이 아니기에 사각형 오브젝트를 탐색하지 못한다. 이러한 경우 사각형 오브젝트는 삼각형 오브젝트를 위협성이 있는 오브젝트로 인식을 하지 못하게 되고 삼각형 오브젝트의 경우는 사각형 오브젝트를 찾지 못한다.

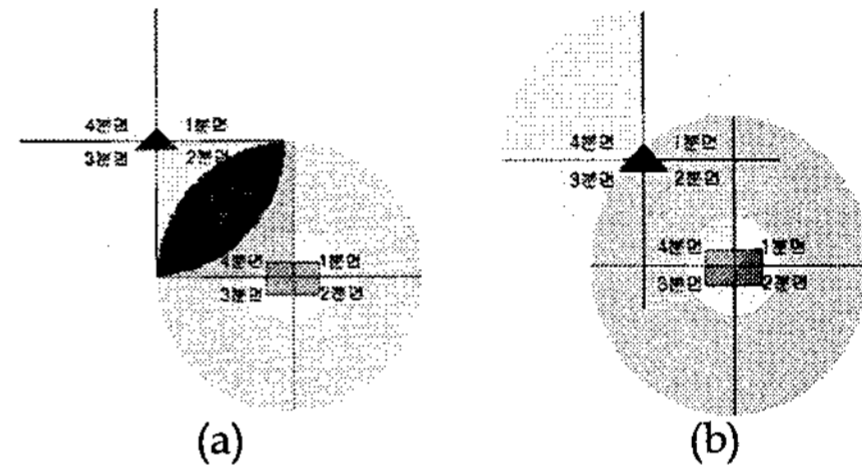


그림 5. 오브젝트 탐색이 불가능 한 경우

그림 5에서 위협성을 인식하는데 필요한 요소들은 두 개의 오브젝트를 일정거리안의 상반되는 분면에 있어야 된다는 것의 조건들이 생긴다. 하지만 모든 오브젝트에 탐색범위를 두게 된다면 아무런 상관도 없는 오브젝트까지 판정을 하게 된다. 하지만 사각형 오브젝트는 B라는 속성만 위협성이 있는 오브젝트로 인식을 하게 된다면 삼각형 오브젝트는 위 조건이 만족되었을 때 속성A에서 속성B로 바뀌게 되어 사각형 오브젝트는 위협성을 인식하게 된다.

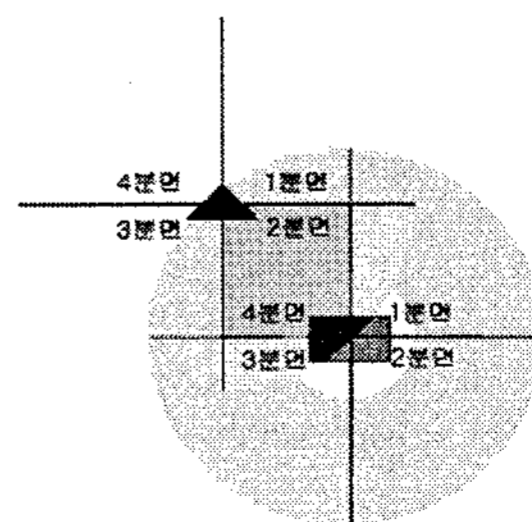


그림 6. 오브젝트 탐색이 가능한 경우

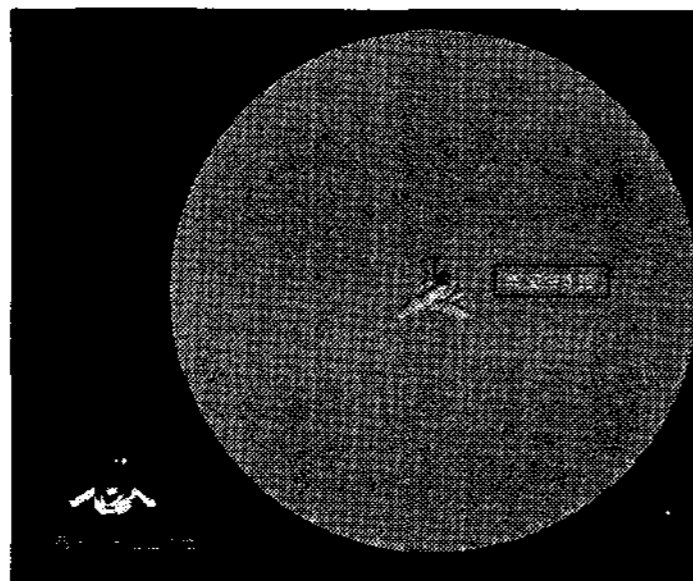
그림6은 모든 조건이 만족 되었을 때 이다. 이것에서 응용을 하게 되면 사각형 오브젝트가 가지는 거리 값을 3단계로 나누고 모든 조건이 만족 되었을 때 삼각형 오브젝트가 다가오게 된다면

사각형 오브젝트와의 거리를 계산하여 위협도를 더욱 자세하게 표현할 수 있다.

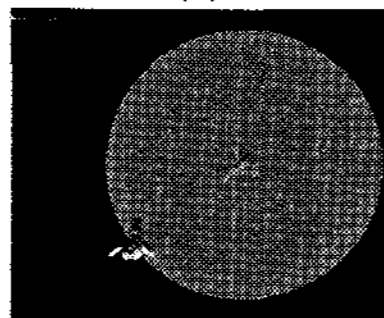
IV. 실험 및 토의

본 논문에서는 변형된 절두체에 플로킹의 회피기법을 응용한 방법을 PC에서 Visual C++[4]로 구현하여 실험하였다. 우선 오브젝트 탐색 실험을 위해 조작 가능한 플레이어 오브젝트와 상대 적 오브젝트를 생성한다.

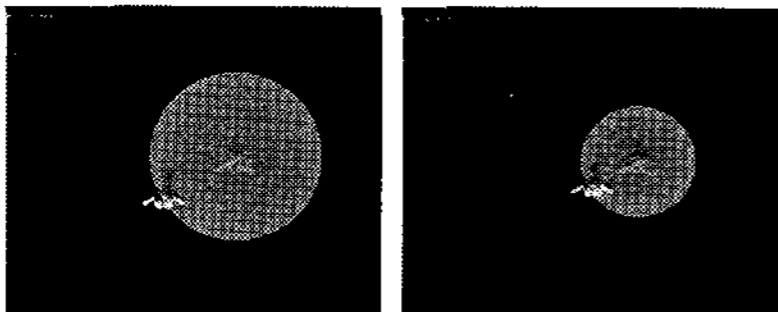
플레이어 오브젝트는 변형된 절두체를 이용하여 주변 적 오브젝트를 항상 판정체크 하도록 하며, 거리에 따라 위협도를 다르게 표시하도록 하였다. 적 오브젝트는 위치값을 0으로 잡아 중앙에 위치하고 회전 값과 속도 값을 주어 제자리에서 회전하며 각 분면을 탐색하도록 한다.



(a)



(b)



(c)

(d)

그림 7. 오브젝트 탐색

그림7은 오브젝트를 탐색하는 과정을 단계적으로 나타내고 있다. 그림에서 각각의 오브젝트에는 MFC와 회피기법이 적용된 상태를 나타내고 있다. 먼저 그림 7.(a)는 적 오브젝트가 플레이어 오브젝트를 인식하지 못하고 있는 상태를 보여주고 있다. 반면 그림 7.(b)는 적 오브젝트가 플레이어 오브젝트를 인식하였지만 일정 범위의 영역에 도달하여 1차 경고(warning-1)를 화면에 나타내는 그림이다. 즉, 적 오브젝트가 플레이어 오브젝트를 인식하게 되는 시작점이 되는 것이다. 그림

7.(c)는 적 오브젝트가 정해놓은 영역에서 더욱 가까이 들어와서 실제 위협성을 내포할 수 있는 영역까지 접근하였으므로 2차 경고(warning-2)를 표현하고 있다. 최종적으로 그림 7.(d)는 플로킹의 회피기법에 의하여 적 오브젝트의 바로 앞에 플레이어 오브젝트가 접근하여 위협도가 고조에 달해 공격성을 띄게 되는 dangers 단계를 나타내고 있다. 이때 적 오브젝트는 완전한 플레이어 오브젝트를 인식하게 되고 공격할 준비를 갖추게 되는 것이다.

이것은 현재 액션 게임에서 기존의 전투방식을 바꿀 수 있는 주요한 전략으로 단순한 액션게임에서 전략을 세우고 게임에 긴장감을 높일 수 있는 중요 요소가 될 것이다.

V. 결론 및 향후 연구

본 논문에서는 변형된 절두체 컬링에 플로킹의 회피기법을 이용하여 액션장르의 게임에서 효과적으로 오브젝트를 탐색할 수 있는 방법을 제안하였다. 제안된 방법을 게임 시뮬레이션에 적용한 결과, 상대 오브젝트를 탐색하고 거리를 계산하여 위협성이 없는 오브젝트는 판정하지 않음으로써 게임의 속도를 올리고 전략을 세우는데 어느 정도 만족 할만한 결과를 얻을 수 있었다. 그러나 조작이 가능한 오브젝트로부터 축이 위에 있는 오브젝트의 경우 4분면의 탐색과 위협도 판정이 느려질 수도 있다. 따라서 변형된 절두체 컬링의 판정범위와 오브젝트의 4분면 판단범위처리 문제를 해결할 수 있는 연구가 계속 되어져야 할 것이다.

참고문헌

- [1] Jin-Hyuk Hong, Sung-Bae Cho, "Evolution of emergent behaviors for shooting game characters in Robocode", Evolutionary Computation, CEC2004, Congress on Vol. 19-23, pp 19-23, 2004
- [2] 최원태, 박창민, "변형된 절두체 컬링을 이용한 3차원 FPS 게임에서의 오브젝트 탐색 연구", 춘계종합학술대회, Vol. 11, NO 1, (사)한국해양정보학회, pp 105-108, 2007
- [3] Mark Deloura, "Game Programming Gems I", 정보문화사, 2005
- [4] 김용준, "3D 게임 프로그래밍", 한빛미디어, 2007
- [5] 주영호, "절두체 컬링 기법과 Quad Tree, LOD 기법을 이용한 렌더링 속도 개선 방법", Technical Report REP05010, 그래픽스 응용 연구실, 부산대학교, 2005