

Navigation on a Terrain

2008년 여름
박경신

Move a Car on a Terrain Model

- 높이가 다른 지형 모델 (terrain model)에서 자동차를 움직이고자 한다면, 우리는 현재 주어진 지형의 위치 (x, y)로부터 지형의 높이 (the height of terrain, 즉 z)를 실시간으로 계산해야 한다.
- osgUtil의 intersection module을 사용하여 이 계산을 수행할 수 있다.
- osg::LineSegment의 intersection detection은 ray를 기반으로 두고 있다.
 - 선분 (line segments)으로부터 ray를 계산한다.
 - 선분의 시작점과 끝점을 osg::Vec3를 사용한다.

2

Intersections

- **hits()** 는 프로그래머가 교차감지 테스트 (Intersection detection test)와 관련된 기본적인 정보를 넘겨준다.
- 하나의 Hit는 scene graph에서 ray가 geometry와 교차할 때 필요한 정보를 준다.
- **osgUtil::IntersectVisitor::HitList**는 scene graph에서 하나의 선분이 교차하는 모든 geometry (또는 같은 geometry를 동시에 여러 번)의 리스트를 넘겨준다.
 - 각 선분마다 교차 감지 테스트가 수행되어 HitList가 생성된다.
 - 이 리스트는 각 선분마다 교차된 것에 대한 Hit 인스턴스를 담고 있다.
 - 만약 교차를 발견하지 못했다면, 이 리스트는 empty이다.

3

Intersections

- osgUtil::IntersectVisitor를 생성하고 적용하여 scene에서 ray와 geometry 간의 교차 감지 테스트가 실행된다.
- 이 클래스는 NodeVisitor에서 파생된 클래스로 NodeVisitor instance와 비슷하게 create과 invoke시킨다.
- 이 선분들마다, 교차 테스트가 수행된 후 히트 리스트 (osgUtil::IntersectVisitor::HitList instance)가 생성된다.

4

Move a Car on a Terrain Model - Process

- 지형 모델에서 자동차를 정확히 움직이게 하기 위하여 다음과 같이 z값 계산을 수행한다.
 - 교차 테스트에 사용하기 위하여 ray의 시작점과 끝점을 Vec3 로 하여 LineSegment instance를 생성한다.
 - IntersectVisitor instance를 생성한다.
 - IntersectVisitor에 LineSegment instance를 추가한다.
 - Scene graph에서 원하는 노드에 대한 IntersectVisitor traversal을 시작한다.
 - 교차 감지 테스트 결과의 world coordinate값을 가져온다 (getWorldIntersectPoint를 사용).

5

Move a Car on a Terrain Model - Implementation

```
double findZPos(double x, double y) { // terrain is a object model node
    osgUtil::IntersectVisitor iv;
    osg::ref_ptr<osg::LineSegment> segDown = new osg::LineSegment;
    const osg::BoundingSphere& bs = terrain->getBound();
    segDown->set(osg::Vec3(x, y, 999),osg::Vec3(x, y, -999));
    iv.addLineSegment(segDown.get());
    terrain->accept(iv);
    if (iv.hits()) {
        osgUtil::IntersectVisitor::HitList& hitList = iv.getHitList(segDown.get());
        if (!hitList.empty()) {
            osg::Vec3 ip = hitList.front().getWorldIntersectPoint();
            printf("z = %lf\n", ip.z());
            return ip.z();
        }
    }
    printf("no hits found\n");
    return 0.0f;
}
```