

네비메쉬를 이용한 3D모바일 레이싱 게임 개발

이병철[○], 안성욱*, 김수균*

[○]*배재대학교 게임공학과

email: kimsk@pcu.ac.kr*

Development of Racing Game using NevMesh Agent

Byeong Cheol Lee[○], Syoungog An*, Soo Kyun Kim*

[○]*Dept. of Game Engineering, Paichai University

● 요약 ●

증가하는 모바일 시장 규모에 맞추어 게임 엔진의 기능이 발전하고 있다. 대형 개발사를 위한 간편한 협업 기능부터 1인 개발자를 위한 간단한 기능들까지 다양한 기능들의 게임 엔진이 사용되고 있으며 그 중 유니티 엔진은 소규모 인디 개발자들을 위한 기능들을 많이 추가하고 또한 발전시켜 많은 개발자들에게 도움을 주고 있다. 본 논문 에서 소개하는 게임은 유니티의 여러 기능을 적극적으로 이용하고, 특히 네비메쉬 에이전트를 사용하여 사용자가 길 찾기 문제를 쉽게 하도록 만든 것이 특징이다.

키워드: Racing Game, Unity3D, NevMesh Agent

I. 서론

유니티의 다양한 게임 개발 기능들과 유니티 에셋 스토어에서 제공하는 여러 종류의 에셋들은 1인 개발자들의 게임 퀄리티를 한층 더 높여주며 활성화된 이 점은 유니티의 저렴한 가격과 맞물려서 다양한 게임이 출시되는데 큰 기여를 하고 있다. 유니티의 다양한 기능들을 습득하는 건 개발자로서 큰 도움이 될 것이란 것은 분명한 사실이다. 본 논문에서는 유니티의 다양한 기능들 중 네비메쉬 에이전트를 활용한 3D모바일 레이싱 개발 방법에 대해 설명한다.

II. 본론

1. 게임의 소개

본 논문에서 만들 게임은 3D모바일 레이싱 게임으로 적AI캐릭터와 경쟁하여 제한시간 동안 상대보다 많은 별을 습득한 쪽이 이기는 게임이다. 버튼을 좌 우로 각각 두 개씩 배치하여 플레이어의 플레이 편의성을 높였다. 또한 카메라를 플레이어의 하위 오브젝트로 배치하여 플레이어를 계속 바라보게 하여 카메라 밖으로 나가지 않도록 하였고 또한 플레이어를 캐릭터가 항상 카메라 중앙에 위치하도록 하였다. 플레이어블 캐릭터가 항상 중앙에 있으므로 플레이어는 자신의 위치와 방해 오브젝트의 위치, 별의 위치를 항상 간편하게 파악할 수 있다.

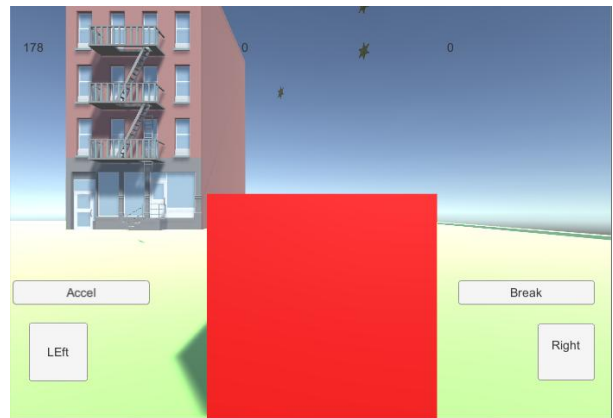


Fig. 1. Play Scene

2. 플레이어

플레이어는 레이싱의 느낌을 주기 위해 엑셀을 누를 경우 가속, 버튼에서 뗄 경우 감속하도록 디자인 하였다. 회전 또한 버튼을 누르고 있을 경우에 회전하게 하여 좀더 쉬운 플레이가 가능하게 하였다. 유니티의 버튼 트리거 기능으로 버튼 눌렀을 때 떴을 때 함수를 따로 설정하여 각각 bool형 변수가 true/false 되게 하였다. go변수가 true가 되면 플레이어가 가속하고 false가 되면 감속한다. 회전은 true일 때 회전 하고 false일때 회전하지 않게 하였다.

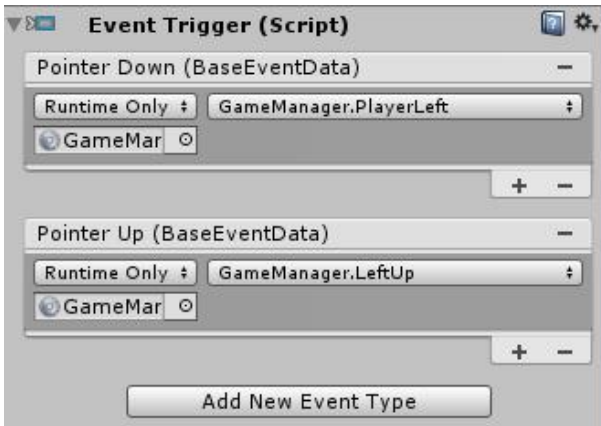


Fig. 2. Button Trigger

3. 네비메쉬 에이전트

네비메쉬 에이전트[1]는 쉽게 말해 길 찾기 기능이다. 오브젝트를 선택하고 오브젝트가 다닐 길과 목적지를 설정하면 오브젝트가 해당 길을 통해 목표지점까지 이동한다. 여기서는 이 네비메쉬 에이전트 기능을 사용하여 적 플레이어가 가장 가까운 별을 찾는 기능 구현에 대해 설명한다.

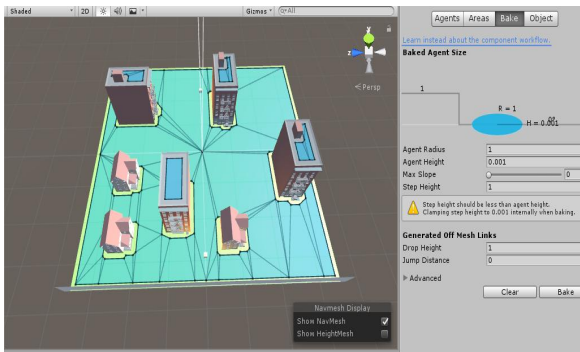


Fig. 3. NevMesh Agent

네비메쉬 에이전트는 길 설정으로 간단하게 사용할 수 있다. 길을 설정한 후 코드에서 목적지만 넣어주면 해당 목적지까지 가는 길을 A*알고리즘을 통해 빠르게 찾아 간다. 이 게임에서는 목적지가 될 별의 포지션과 그 포지션까지의 거리를 배열로 설정하여 에너지 오브젝트에 저장하였다. 이후 목적지 별과 충돌할 때마다 목적지를 다시 초기화하도록 하였다. 목적지 초기화는 플레이어와 별의 충돌시에도 넣어 주어 AI가 노리던 오브젝트를 플레이어가 먼저 습득 할 경우 다시 길을 찾을 수 있게끔 하였다.

목적지를 찾을 때는 이중 for문을 사용하였다. 첫 번째 for문에서는 제일 가까운 오브젝트를 찾았고 두 번째 for문에서는 첫 번째 for문에서 찾은 오브젝트가 true 인지 false인지 검사하였다. 우선 for문을 돌리기 전 별의 번호와 거리를 받을 변수를 선언하였고 첫 번째 별부터 마지막 별 까지 for문을 돌려 그 번호 별의 거리와 저장된 최소거리를 비교하고 더 짧으면 두 번째 for 문으로 들어갔다. 두 번째 for문에서 해당 별이 true인 것을 확인하면 선언한 변수에 해당

값을 대입 하였다. 이렇게 마지막 별까지 확인하여 최종적으로 결정된 별의 좌표를 길찾기에 넣어서 길을 찾게끔 하였다.

4. 아이템

본 게임에서 등장하는 아이템은 별 한 종류로 플레이어와 AI가 서로 경쟁하며 습득할 아이템이다. 게임이 시작되면 X, Z축을 랜덤하게 결정하여 위치를 설정하였고 Y 20값에서부터 떨어지게 하여 역동적인 느낌을 주었다. 만일 Map 태그를 가진 오브젝트와 부딪힐 경우 다시 X, Z축을 설정하게 하여 플레이어와 적이 갈 수 없는 지형에서 생성되는 것을 방지하였다. 아이템은 충돌[2]시 플레이어 또는 AI의 하위 오브젝트로 설치된 파티클을 true로 바꾸고 사운드를 재생시켜 충돌했다는 느낌을 주게 하였다.



Fig. 4. Item Image

본 논문에서는 다음과 같은 개발환경에서 게임을 제작한다. Window 10, intel(R) core(tm)i5-7200u cpu, 8.00GB Unity 2018.2.6f1 버전에서 가능한 기능들을 사용하였다.

III. 결론

본 논문에서는 유니티의 기능들 중 네비메쉬 에이전트를 적극적으로 활용한 3D모바일 레이싱게임 구현에 대하여 설명하였다. 이처럼 유니티의 기능을 적극적으로 활용하면 쉽고 간단하게 게임을 만들 수 있다. 본 논문에서는 네비메쉬 에이전트를 통한 길 찾기를 쉽게 구현하는 법을 설명하였다.

REFERENCES

- [1] <https://docs.unity3d.com>
- [2] <http://www.devkorea.co.kr>