|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | | **작성자** | **2019180049 윤우영**  **2019180046 김강휘**  **2016180015 김지호** | **팀명** | Survil |
| **주차** | **7** | **기간** | **2022.02.13~2022.02.19** | **지도교수** | **정 내 훈**(서명) |
| **이번주 한일** | * **윤우영: 불필요 Memcpy 삭제 후 성능 향상, ShadowMap, TemperatureMap** * **김강휘: citizen spawn 오류 수정, citizen blueprint -> citizen c++ 작업** * **김지호:** | | | | |

**<상세 수행내용>**

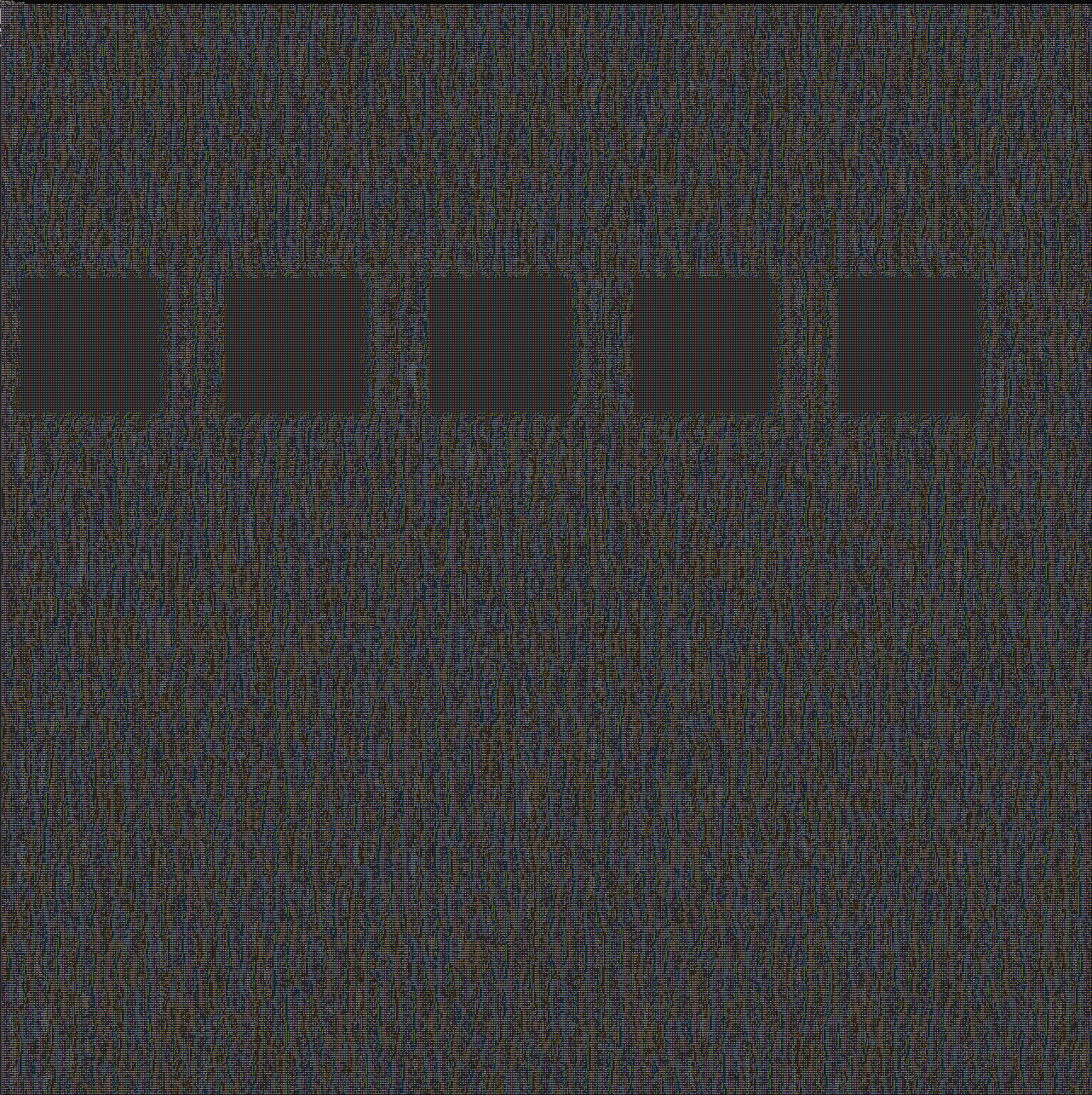
윤우영:

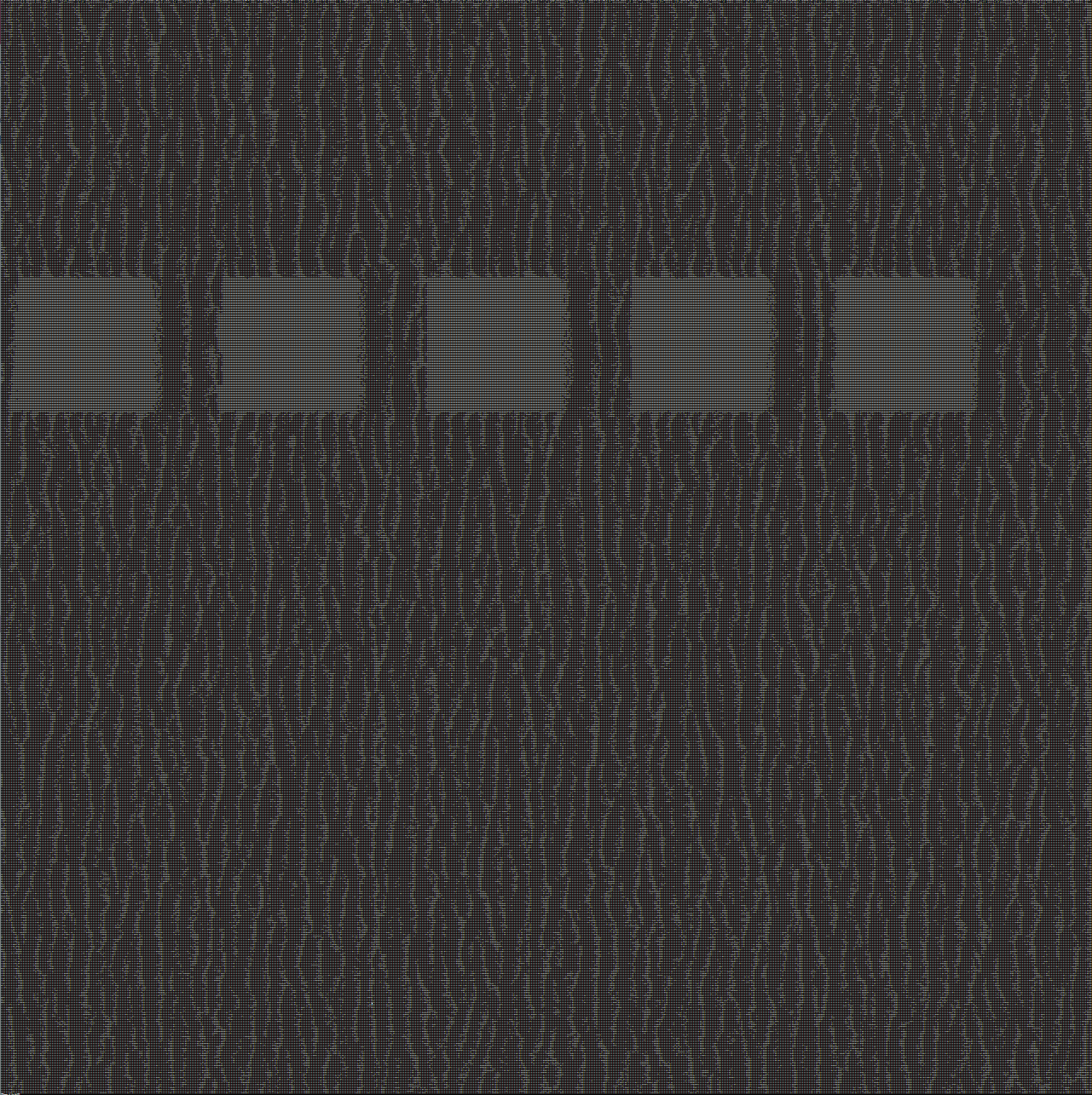
39936\*39936에서 32000\*32000으로 변경했더니 한 루프에 약 5초에서 약 4초로 줄어들었다.

Shadow\_map과, Temperature\_map을 추가했더니 메모리 사용량이 약 8.7GB로 늘어났다.

Shadow\_map을 업데이트 하는 함수를 제작했다. 아래는 지형, Shadow\_map 비교 사진.

태양 각도는 60도 일 때다.





Cuda Kernel을 호출하고 난 뒤에 항상 Device to Host Memcopy를 하는 실수가 있었다… 연속된 Cuda Kernel 호출에는 DeviceSyncronize만 하고 루프의 마지막에만 Memcpy를 하게끔 변경해서 바람 한번 부는데 (32000\*32000) 2.0 ~ 2.3초가 걸리게 했다. 거의 절반으로 줄었다.

ShadowMap과 TemperatureMap을 만들어서 Cuda함수를 제작했다.

아래 이미지는 태양 각도 동쪽 25도일때의 모습이다.

TerrainMap

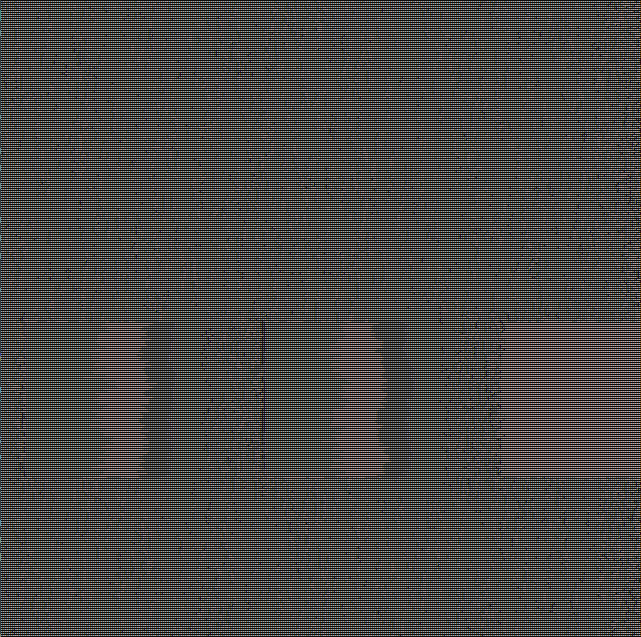
실외, 스피커, 바둑판식, 창살이(가) 표시된 사진

자동 생성된 설명

ShadowMap



TemperatureMap

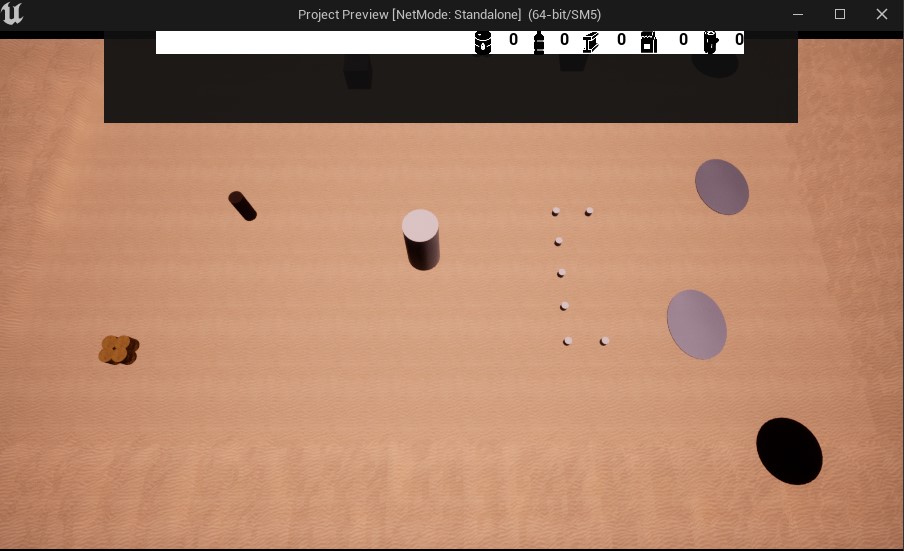


TemperatureMap에서 그림자가 있으면 온도가 떨어지고, 그림자가 없으면 온도가 올라가게끔 했다.

온도를 얼마나 낮추고 얼마나 올려야 하는지 공식을 만들어야 한다. 하루 시간, 태양 각도 업데이트 횟수, 등 고려할 것이 많다.

김강휘:

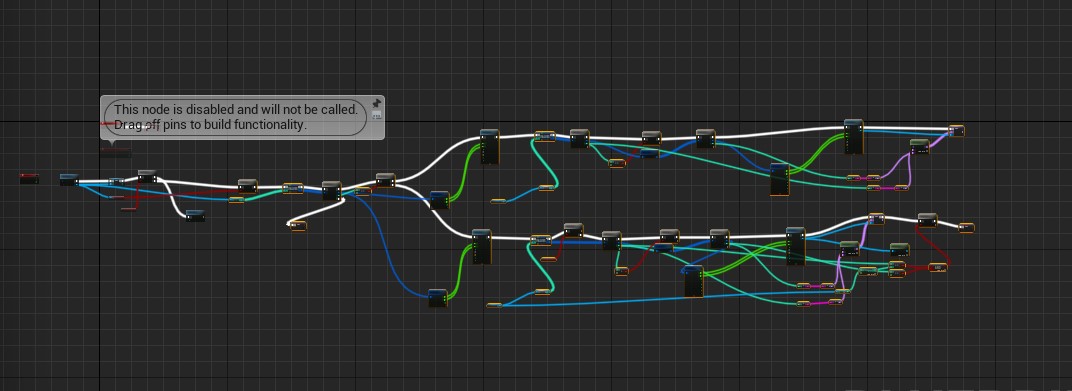
Citizen spawn fix

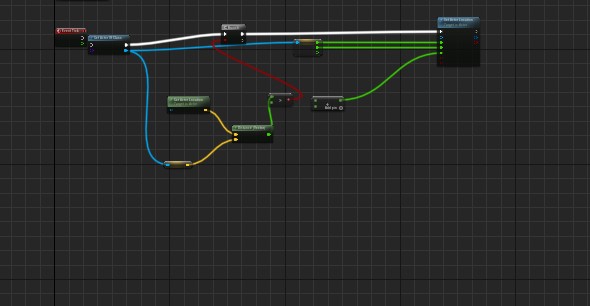


Citizen이 간헐적으로 각자 위치에 소환되지 않던 오류를 수정하였다.

(Recv를 받기전에 citizen이 스폰되어 0,0,0,좌표로 생성되었던 오류)

Citizen을 Blueprint로 작업하던 사항을 코드로 구현화하였다.





코드 중 일부 캡쳐



blueprint에서 서버와 연결해서 작업하면 복잡하고 코드를 남이 고치기가 어려워서

c++로 구현하기로 하였다.

C++을 통해 server와 좀 더 쉽게 연동이 가능하게 되었고

불필요한 데이터 복사 작업이 필요 없어졌다.

김지호:

|  |  |  |  |
| --- | --- | --- | --- |
| **문제점 정리** |  | **해결 방안** |  |
| **다음 주차** | **8** | **다음 기간** | **2022.02.20~2022.02.26** |
| **다음주 할 일** | 윤우영:  김강휘: **resource blueprint -> resource c++ 작업, resource UI 추가**  김지호: | | |
| **지도교수**  **Comment** |  | | |