유니티 엔진을 이용한

의료 영상 가시화 프로그램

Proposal

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 팀 명 | 상밀레 | | | |
| 담당 교수 | 계희원 교수님 | | | |
| 팀 원 | 이름 | 학번 | 전화번호 | 메일 |
| 이상은Ο | 1692055 | 01051707327 | [weslse@hansung.ac.kr](mailto:weslse@hansung.ac.kr) |
| 김 청 | 1392013 | 01096280718 | [kkseoul\_0@naver.com](mailto:kkseoul_0@naver.com) |
| 조예원 | 1491036 | 01094551104 | [nov4com@naver.com](mailto:nov4com@naver.com) |
| 최낙수 | 1492039 | 01037903411 | [combrain817@naver.com](mailto:combrain817@naver.com) |
| Project URL | <https://github.com/gongmillehansung/gongmille> | | | |

목차

1. 프로젝트 수행 목적
   1. 프로젝트 정의
   2. 프로젝트 수행 동기 및 목표
2. 프로젝트 개요
   1. 프로젝트 기능 설명
   2. 프로젝트 구조
   3. 프로젝트의 제약 조건
   4. 개발 도구 및 관련 기술
      1. 개발 도구
      2. 관련 기술
3. 결과로써 제출할 실적물 목록
4. 프로젝트 수행 능력 향상을 위한 체계 및 일정
   1. 조직도
   2. 마일스톤 및 일정
   3. 프로젝트 수행 일정 (주 단위)
5. 참고자료
6. 프로젝트 수행 목적
   1. 프로젝트 정의

유니티 엔진을 이용하여 실제 볼륨 데이터(CT 등의 의료 영상)에 대해 볼륨 렌더링을 수행하고 해당 볼륨 데이터를 조작하는 프로그램

* 1. 프로젝트 수행 동기 및 목표

대개 의사가 환자의 CT 등의 의료 데이터를 보통 2d 단면 이미지로 관찰하고 해석, 환자에게 진단 결과를 알려주는 절차를 거친다. 마지막 단계인 환자에게 진단 결과를 알려주는 경우 2D 단면 이미지를 보여주고, 앞에 놓여진 모형 등을 가리키며 설명하는 경우가 많다. 이 때 환자가 해당 환부에 대한 지식이 부족하거나 설명하기 어려운 부위인 경우, 환부 위치에 대해 제대로 설명하기 어려운 경우가 생긴다. 이 프로젝트의 결과물을 통해 설명의 용이성을 얻는다.

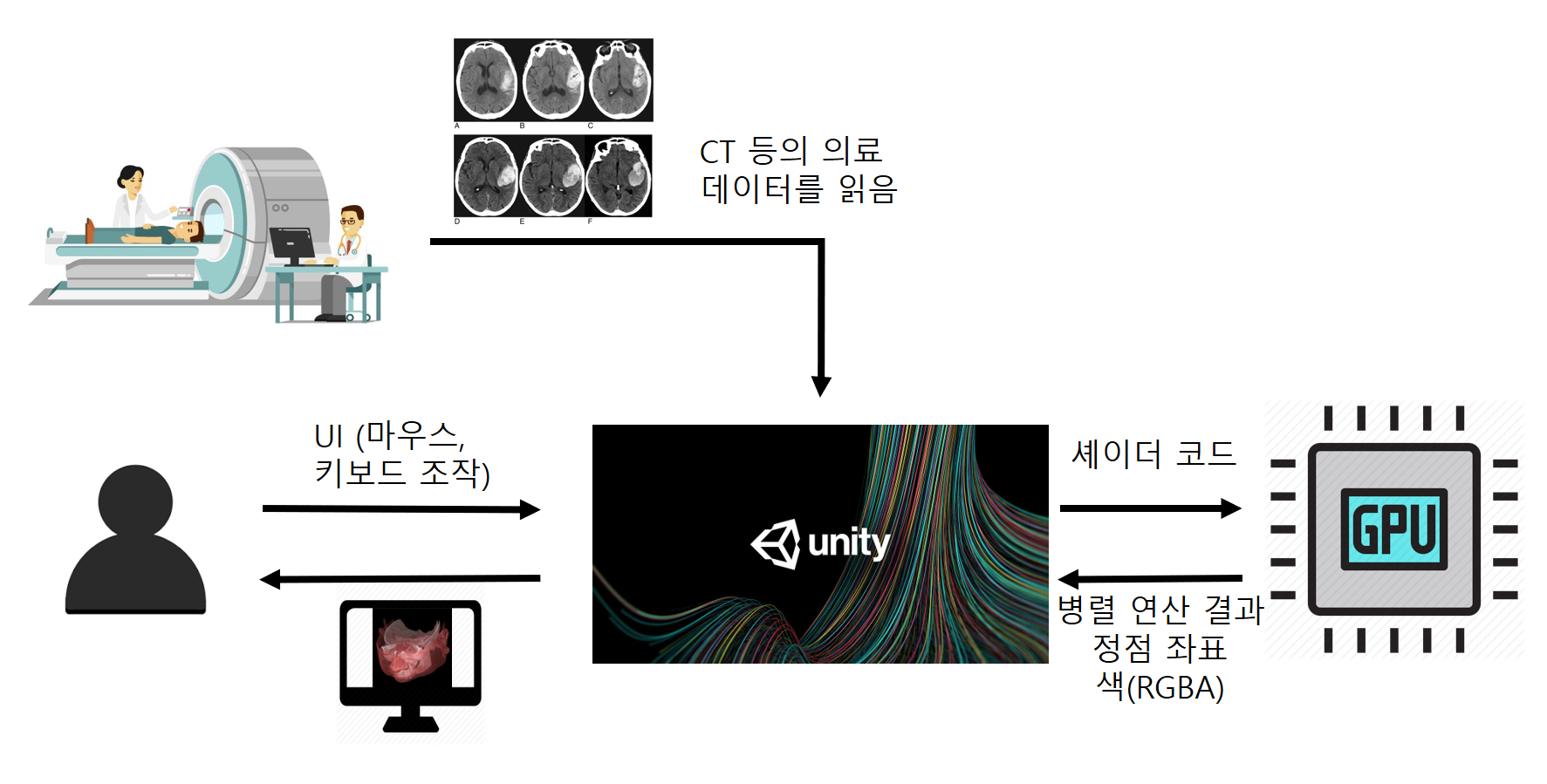
한편, 수술은 한 번의 실수로도 환자의 생명이 달릴 수 있는 굉장히 고위험의 작업이다. 실제 의료 데이터를 이용하여 가상 수술 시뮬레이터를 만들어 다량의 연습을 하게 되면 실전 상황에서도 어렵지 않고 위험하지 않게 해낼 수 있을 것이다. 이 프로젝트에서 GPU를 이용한 고속 연산을 이용하여 가상 수술 시뮬레이터로 발전할 수 있는 바탕을 다지고자 한다.

1. 프로젝트 개요
   1. 프로젝트 기능 설명

이 프로젝트는 의료 데이터를 바이너리 파일로 파일 입출력을 이용하여 유니티 엔진에서 읽는다. 유니티 엔진에서 제공하는 셰이더(Shader)들을 이용하여 다양한 그래픽 연산들을 GPU를 이용하여 빠르게 처리한다. 셰이더에서 처리한 텍스처(Texture)를 유니티 기본 도형인 큐브(Cube)에 렌더링한다.

사용자는 마우스와 키보드를 이용하여 카메라, 관찰 범위, 전이 함수(Transfer Function) 등을 조절하여 원하는 정도, 범위 등을 볼 수 있다.

* 1. 프로젝트 구조



* 1. 프로젝트 기대효과

대개 의사가 환자의 CT 등의 의료 데이터를 보통 2d 단면 이미지로 관찰하고 해석, 환자에게 진단 결과를 알려주는 절차를 거친다. 마지막 단계인 환자에게 진단 결과를 알려주는 경우 2D 단면 이미지를 보여주고, 앞에 놓여진 모형 등을 가리키며 설명하는 경우가 많다. 이러한 상황에서 스캔한 환자의 이미지 등을 3D로 보여주며 설명한다면 이전의 방식보다 설명에 용이할 뿐만 아니라 설명을 들어야하는 입장에서도 이해하기 쉬워질 것이다.

* 1. 프로젝트의 제약 조건  
     **- 경제성**

사용자의 프로그램 실행 환경(CPU, GPU, etc.)에 따라 연산 속도가 결정되며, 렌더링 속도가 크게 다를 수 있음

**- 미학**

UI, UX 부분에 있어서 프로그램의 사용자가 편리하게, 어렵지 않게 느낄 수 있도록 구성한다.

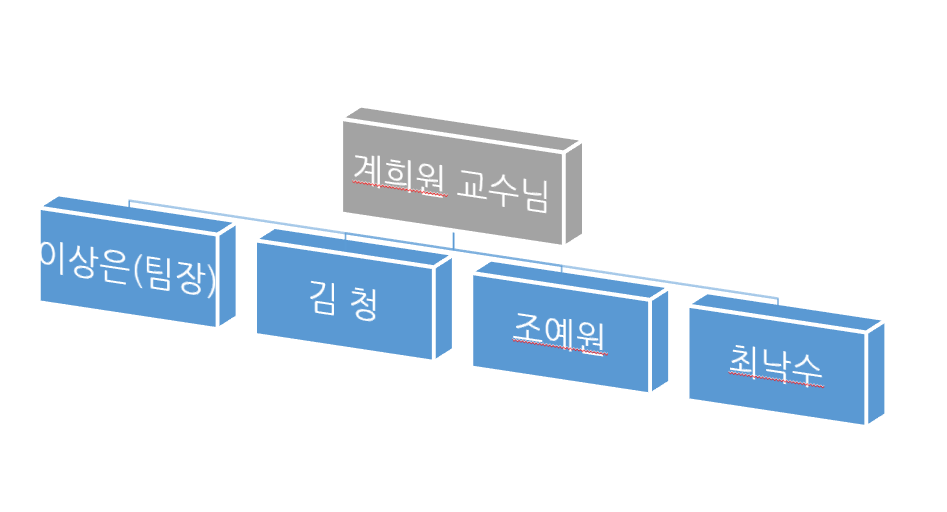
* 1. 개발 도구 및 관련 기술

|  |  |
| --- | --- |
| 구분 | 설명 |
| Visual Studio | 마이크로소프트 비주얼 스튜디오(Microsoft Visual Studio)는 마이크로소프트 윈도우에서 작동하며, 다양한 언어로 프로그래밍할 수 있는 마이크로소프트의 통합 개발 환경이다. 윈도우 OS에서 가장 강력한 IDE 중에 하나로 손꼽힌다. |
| Unity Engine | 유니티([영어](https://ko.wikipedia.org/wiki/%EC%98%81%EC%96%B4): Unity)는 3D 및 2D [비디오 게임](https://ko.wikipedia.org/wiki/%EB%B9%84%EB%94%94%EC%98%A4_%EA%B2%8C%EC%9E%84)의 개발 환경을 제공하는 [게임 엔진](https://ko.wikipedia.org/wiki/%EA%B2%8C%EC%9E%84_%EC%97%94%EC%A7%84)이자, 3D 애니메이션과 건축 시각화, [가상현실](https://ko.wikipedia.org/wiki/%EA%B0%80%EC%83%81%ED%98%84%EC%8B%A4)(VR) 등 인터랙티브 콘텐츠 제작을 위한 통합 저작 도구이다. [2016년](https://ko.wikipedia.org/wiki/2016%EB%85%84) 기준으로, 세계 게임엔진 시장의 45%를 차지하는 가장 인기있는 게임엔진이다. 엔진 자체에 라이트 매핑, [물리 엔진](https://ko.wikipedia.org/wiki/%EB%AC%BC%EB%A6%AC_%EC%97%94%EC%A7%84) 등 [미들웨어](https://ko.wikipedia.org/wiki/%EB%AF%B8%EB%93%A4%EC%9B%A8%EC%96%B4)를 탑재했으며, 에디터에 내장된 애셋 스토어를 통해 다양한 기능의 애셋을 다운로드하여 사용할 수 있다. |

* + 1. 개발 도구
    2. 관련 기술

|  |  |
| --- | --- |
| 구분 | 설명 |
| Shader | 컴퓨터 그래픽스 분야에서 셰이더(shader)는 소프트웨어 명령의 집합으로 주로 그래픽 하드웨어의 렌더링 효과를 계산하는 데 쓰인다. 셰이더는 그래픽 처리 장치(GPU)의 프로그래밍이 가능한 렌더링 파이프라인을 프로그래밍하는 데 쓰인다. 셰이더는 표면상으로 무한해 보이는 효과를 만들기 위해 영화 후처리, CGI, 비디오 게임에 널리 쓰인다. 이 프로젝트에서는 셰이더 언어인 HLSL을 이용한다. |
| Volume Rendering | 과학적 시각화와 컴퓨터 그래픽스 분야에서, 볼륨 렌더링은 3차원 스칼라 장 형태의 이산 샘플링 데이터를 2차원 투시로 보여주는 기술을 말한다. 전형적인 3차원 데이터는 CT, MRI, 혹은 MicroCT 스캐너에서 요구되는 2차원 단면 이미지 그룹을 말한다. 또한 해당 기술은 복셀이라는 최소 단위에 색상과 불투명도(RGBA) 값 등을 정의하여 렌더링 시에 색상을 가지고 이미지를 사실적으로 보여낸다. |

1. 결과로써 제출할 실적물 목록
2. 프로젝트 수행 능력 향상을 위한 체계 및 일정
   1. 조직도



* 1. 마일스톤 및 일정
  2. 프로젝트 수행 일정 (주 단위)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 볼륨 렌더링 | | 3월 | | | | | | | | 4월 | | | | | | | | 5월 | | | | | | | |
| 작업 구분 | | 1주차 | | 2주차 | | 3주차 | | 4주차 | | 1주차 | | 2주차 | | 3주차 | | 4주차 | | 1주차 | | 2주차 | | 3주차 | | 4주차 | |
| 기획 | 자료조사 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 기술선정 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 제안서 작성/수정 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 학습  및  정리 | C# |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Unity |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| HLSL |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Graphics |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 설계 | UI |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Volume  Rendering |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 구현 | UI |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Volume  Rendering |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 테스트 | 프로그램 보완 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 오류 검출 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 테스트 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 유지보수 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

1. 참고자료

Wikipedia, Unity3d document