

AR 기술을 활용한 소셜 네트워킹 서비스 개발 착수 보고서



메아리(meARy)

201924447 김진영

201924557 임석윤

201924613 허취원

목차

1. 연구 배경 및 목표
 - a. 연구 배경
 - b. 과제 목표
2. 요구조건 분석 및 설계
 - a. 조건 분석
 - b. 개발 환경
 - c. 전체 구상도
3. 연구 방향
 - a. 동영상 기반 3D 모델 생성
 - b. 3D 모델을 humanoid rig로 관절화
 - c. 포스팅 사진에서 관절정보 추출
 - d. 3D 모델에 관절정보 적용
 - e. GPS 기반 모델 AR로 표현
4. 제약사항
 - a. 3D 모델링 위한 고성능 GPU 필요
 - b. 3D 오브젝트 포즈 자연스럽게 재현 필요
 - c. GPS 한계로 인한 오브젝트 정확한 위치 지정 어려움
 - d. 주변 환경과 3D 오브젝트의 조화
5. 개발 일정 및 역할 분담
 - a. 개발 일정
 - b. 역할 분담
6. 참고문헌

1. 연구 배경 및 목표

a. 연구 배경

21세기 스마트폰의 보급이 보편화 되면서 소셜 네트워킹 서비스의 사용이 많이 증가하였다. 기존 서비스들은 사진, 동영상, 텍스트 위주의 기록을 남긴다. 이런 평면적인 기록은 타인과 추억을 공유하기에 그친다. 우리는 AR 기술을 사용해 기록을 입체적으로 시각화하여 사용자들에게 기록과 상호작용하는 경험을 선사하고자 한다.

b. 과제 목표

사용자의 영상을 기반으로 3D 모델링을 수행한다. 이후 올린 포스팅에서 사용자를 인식하여 이를 3D 모델로 재현한다.

2. 요구조건 분석 및 설계

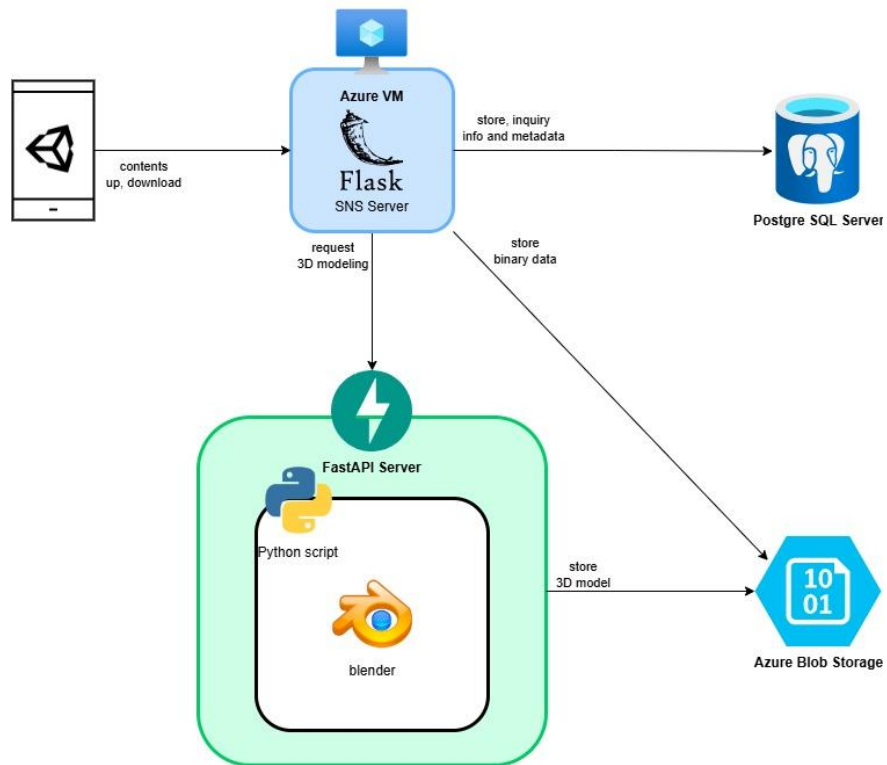
a. 조건 분석

- i. 플랫폼 : iOS, Android 양쪽 지원
- ii. 사용자 기능 :
 - ① 영상 업로드 및 3D 모델 생성 요청
 - ② 포스트 생성 (위치 정보 포함)
 - ③ 다른 사용자의 AR 포스트 보기
- iii. 서버 기능:
 - ① 영상 처리 및 3D 모델링
 - ② 사용자 정보, 포스트, 모델 저장
 - ③ 요청 처리 API 제공 (Flask 기반)
- iv. AR 기능:
 - ① Unity 기반 앱 개발 (AR Foundation 사용)

b. 개발 환경

- i. Unity
- ii. AR Foundation
- iii. Python
- iv. PostgreSQL
- v. Linux
- vi. Azure VM
- vii. Azure Blob Storage
- viii. Flask
- ix. FASTAPI
- x. Blender
- xi. Android
- xii. iOS

c. 전체 구상도



- 콘텐츠 포스팅 시나리오
 1. 사용자의 3D Model을 생성하기 위한 360도 영상(360 orbit shot)을 찍어 upload
 2. 3D Modeling 서버 측에서 동영상을 기반으로 3D Model 생성
 3. 사용자가 사진 포스팅을 올림
 4. SNS 서버 측에서 GPS 정보와 사진 속의 사람을 식별하고, 관절 정보 계산 후 DB에 저장
- 포스트 AR 재현 시나리오
 1. 사용자가 특정 장소에서 서비스를 호출
 2. SNS 서버 측에서 사용자의 GPS 위치를 기반으로 50미터 내에서 올라온 포스팅 조회
 3. 조회된 포스팅의 3D 모델, 관절 정보, 위치를 활용해 AR로 표현

3. 연구 방향

- a. 동영상 기반 3D 모델 생성
 - i. 사용자가 360 Orbit shot을 촬영하고, 그 영상에서 FFmpeg를 사용해 이미지 프레임을 추출한다.
 - ii. 추출한 30~50장의 이미지로 3DF를 이용하여 영상 속 인물 3D 모델링한다.
- b. 3D 모델을 humanoid rig로 관절화
 - i. 3DF에서 만든 3D 모델에 Blender Rigify를 이용하여 뼈대를 만든다.
 - ii. MediaPipe 에서 얻은 2D 관절 위치를 Blender 카메라 뷰와 정렬하고 3D 관절 위치를 Retargeting 하는 것을 Python 스크립트로 자동화 한다.
- c. 포스팅 사진에서 관절정보 추출
 - i. 사용자가 올린 포스팅 사진을 자세 추정 모델에 업로드한다.
 - ii. MediaPipe Pose 를 사용해 33개의 관절 정보(x, y, z) 추출한다.
 - iii. MediaPipe는 2D 이미지로부터 3D 관절 정보를 추출하기에 이때 제공되는 상대적인 깊이 값(z)의 정확도가 관건이다. 깊이 값이 부정확할 시 단일 이미지 기반의 Depth Estimation 기법인 다른 모델(ex. MiDas) 추가 도입 방안을 고려하고 있다.
- d. 3D 모델에 관절정보 적용
 - i. MediaPipe Pose로부터 추출한 3D WorldLandmark 정보를 Unity의 Humanoid Rigging 모델에 적용한다.
- e. GPS 기반 모델 AR로 표현
 - i. AR 앱에서 카메라의 위치, 방향, 환경매핑을 활성화한다.
 - ii. Unity Occlusion Manager를 사용해 피사체의 거리를 찾는다.
 - iii. GPS와 카메라 각도, 피사체의 거리를 기반으로 3D 모델을 Google Geospatial API에 anchor로 등록하여 AR에 표현한다.

4. 제약사항

- a. 3D 모델링 위한 고성능 GPU 필요

360도 영상을 기반으로 3D 모델 생성에는 컴퓨팅 자원이 많이 요구된다. CPU로 생성하는 경우 약 20분 이상이 소요될 수 있다. 그렇기에 고성능 GPU 사용이 불가피하다.
- b. 3D 오브젝트 포즈 자연스럽게 재현 필요

MediaPipe 라이브러리를 통해 2D 사진 내 피사체의 자세를 인식해 3D 모델에 적용할 예정이다. 하지만 MediaPipe의 한계로 정확한 자세를 추정하기 어렵기에 해당 자세가 적용될 3D 모델이 부자연스러워질 수 있다.
- c. GPS 한계로 인한 오브젝트 정확한 위치 지정 어려움

GPS는 오차가 약 5m이하로 발생할 수 있다. 이로 인하여 의도하지 않은 위치에 3D 모델이 배치될 수 있다. 이 한계를 해결하기 위해 Google Geospatial API를 활용할 예정이다. 해당 API는 Google Maps Platform의 데이터 소스를 통해 배경 유사도를 계산하여 GPS만 사용하는 것보다 정확한 위치를 계산할 수 있다.
- d. 주변 환경과 3D 오브젝트의 조화

AR 서비스의 특성 상 사진 속 피사체가 주변 환경과 상호작용하고 있는 경우 재현할 시 시각적 이질감이 느껴질 수 있다.

5. 개발 일정 및 역할 분담

a. 개발 일정

개발구분	세부항목	5	6	7	8	9	10
기획	기획 및 주제 고도화						
	사전 조사 및 개발 환경 구축						
핵심 기능 개발	3D 모델링 및 리깅						
	MediaPipe 기반 관절 추출						
	3D 모델 생성 파이프라인 개발						
Azure DB, 스토리지 구축	DB, 스토리지 설계 및 구축						
Unity 개발	Unity 앱 개발						
서버 개발	Flask, FastAPI 서버 개발						
	서버 앱 연동						
테스트 및 배포	보완사항 수정 및 배포						

b. 역할 분담

이름	담당
김진영	Unity 기반 AR 앱 개발, SNS 서버 개발, Blob Storage 구축
임석윤	Unity 기반 AR 앱 개발, 3D Object Rigging, 3D Modeling 서버 개발
허취원	Azure Cloud 구축, SNS 서버 개발, DB 연동

6. 참고문헌

1. Google MediaPipe Documentation
https://ai.google.dev/edge/mediapipe/solutions/vision/pose_landmarker/python?hl=ko
2. Unity AR Foundation Documentation
<https://docs.unity3d.com/Manual/index.html>
3. PostgreSQL Official Documentation
<https://www.postgresql.org/docs/>
4. Google Geospatial API Documentation
<https://developers.google.com/ar/develop/geospatial?hl=ko>