**프로젝트 최종 제안서**

**1. 팀원**

<B711225 성민재>, <B911400 박수민>, <B911405 윤태인>

**2. 기획**

많은 사람들이 관절, 척추관련 질환을 겪으면서도 원인 중에 올바르지 않은 걸음걸이가 있다는 사실을 알지 못합니다. 그만큼 걸음걸이에 대한 관심과 정보가 부족하다는 것을 알 수 있습니다.

**·** 기존 현행

잘못된 걸음걸이로 인한 통증 발생 -> 통증 관련 병원 방문 -> 검사 후 치료 진행

이렇게 될 경우 병원 치료 및 자세 교정에 있어 비용이 발생하고 추가적으로 교정을 위한 디바이스 필요하고 직접 병원에 방문하여야 하는 불편함이 있습니다.

**3. 프로젝트 주제**

**·** 이전 데이터와의 비교 분석으로 장기간에 걸친 교정 서비스 제공

-> 피트니스 분석처럼 잘못된 자세에 대한 단순 자세 교정 목적이 아닌 습관의 교정

**·** 단순히 걸음걸이의 옳고 그름이 아닌 관절과 관련된 질병정보 제공에 중점

-> 정보를 통해 질병 예방 효과를 누릴 수 있다.

**·** 금전적 부담을 덜고 편리한 접근성 제공을 취지로 함

-> 현대인의 바쁜 일상 중에 시간과 돈을 절약할 수 있다.

**사용 방법**

  - 사용자가 웹을 통해서 동영상을 서버로 업로드 하면 분석 후 결과를 Web으로 도출

**결과 도출**

**·** 분석결과 제공 (걸음걸이 간 문제점 및 걸음걸이 종류 포함)

**·** 같은 걸음걸이를 유지 시에 오는 질병 종류 안내

**·** Animation model

-> 올바른 걸음걸이를 애니메이션을 통한 직관적 표현

-> 사용자의 걸음걸이 문제점을 올바른 걸음걸이와 비교하여 시각화 제공

  -> 새로운 결과에 대해 이전 결과와 비교 분석 제공

**4. 역할 분담**

**· 성민재**

- My SQL을 통해 Data Base 설계

- My SQL과 Django 연동 및 기능(회원가입, 로그인, 업로드 된 동영상 저장) 구현

- Open Pose 라이브러리를 활용해 Pose Estimation 구현

- Coco Dataset을 사용한 AI 학습 후 결과 도출

**· 박수민**

- Unity와 Django 연동 및 Animation 환경 셋팅

- Unity Animation Model Asset 올바른 걸음걸이 동작 수정

- Open Pose 라이브러리를 활용해 Pose Estimation 구현

- Coco Dataset을 사용한 AI 학습 후 걸음걸이 결과 도출

**· 윤태인**

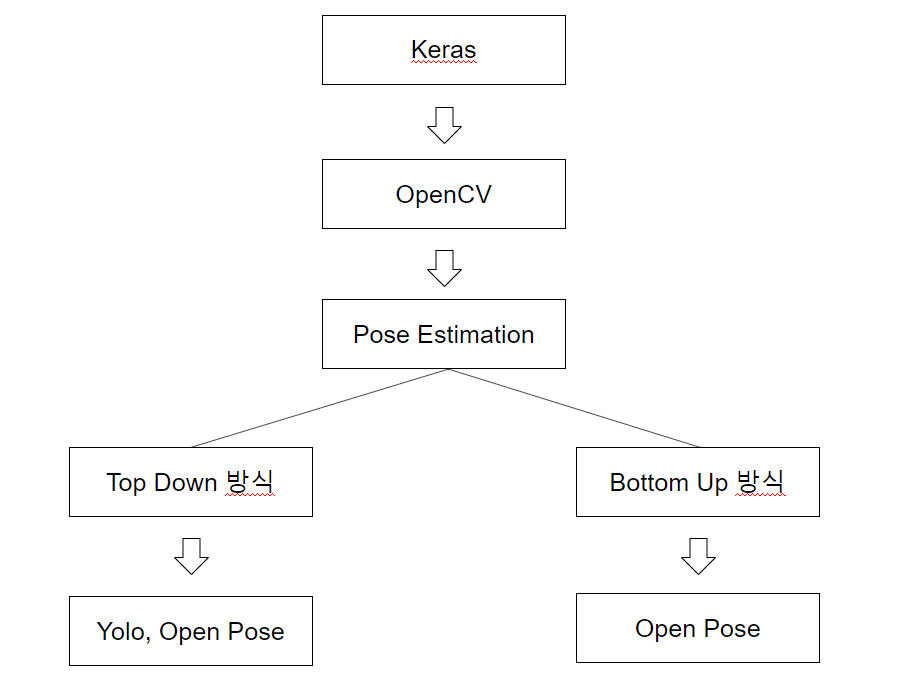
- React(주요기능별 Component구축, Redux를 이용한 상태관리시스템 구축)

- React와 Django 연동을 통해 구현된 기능 활성화 (회원가입, 로그인, 동영상 업로드)

- Web Application에 Unity Animation Model 시각화

**5. 기술적 배경**

**-Pose Estimation**



Pose estimation이란 사람의 신체 관절인 KeyPoints가 어떻게 구성되어 있는지 위치를 측정(Localization)하고 추정(Estimation)하는 문제입니다. 컴퓨터 비전에는 이 밖에도 여러 분야가 속해 있지만 저희 프로젝트의 가장 큰 요인은 사용자의 걸음걸이를 분석하는 것이기 때문에 Pose estimation을 사용하기로 하였으며, 이는 크게 두 가지 Top Down 방식과 Bottom Up방식으로 나뉩니다.

**· Top Down방식**

**:** 영상이나 이미지에서 사람을 Detection하고, Bounding Box 내부에서 포즈를 추정하는 방식입니다. 예를 들어, Yolo 알고리즘을 사용해 객체 탐지를 한 뒤 OpenPose를 활용하여 Bounding Box 내부에서 관절을 Key Points로 나타내어 Skeleton 처리를 하게 됩니다.

\*한계점

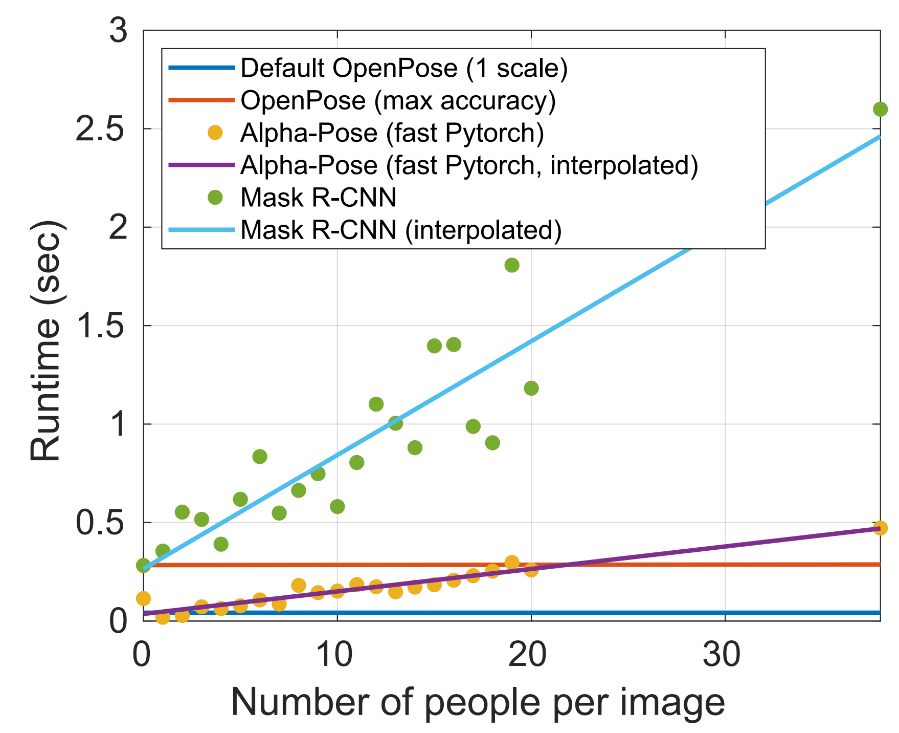
: Detection이 항상 선행되어야 하므로 시간이 오래 걸리며, 사람이 다수일 경우 Detection된 사람마다 포즈를 추정하기 때문에 느립니다.

**· Bottom Up방식**

**:** 영상이나 이미지에 포함된 사람의 KeyPoints를 모두 추정하고 KeyPoints간의 상관관계를 분석하여 포즈를 추정하는 방식입니다.

저희는 한 사람의 걸음걸이 동영상을 분석할 것이므로, 객체 탐지의 필요성을 찾지 못했습니다. 또한 Bottom up 방식 중 가장 일정한 런타임을 가지며 효율적인 모습을 보여주는 **OpenPose**를 사용하여 **Bottom Up** 방식으로 Pose Estimation을 하기로 결정하였습니다.

**- 런타임 비교**



**- 데이터셋**

Pose estimation에서 사람의 자세를 추정하기 위해 어떤 데이터셋을 사용함에 따라 관절을 나타내는 KeyPoint와 더불어 성능도 달라집니다. 대표적인 데이터셋으로 MPII, COCO, LSP 등이 있습니다.

**· MPII**

: 15개의 Key Points가 있으며 약 4만 명의 인물이 포함된 2만5천 장의 이미지로 구성되어 있습니다. 각 이미지에는 관절 좌표뿐 아니라 신체 부분 폐색 등 410개의 활동 레이블링이 제공됩니다.

**· COCO**

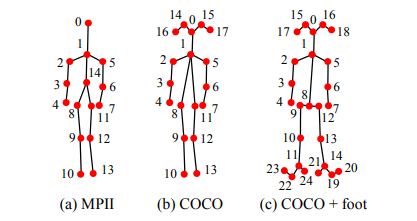
: 18개의 Key Points가 있으며 32만5천 장의 이미지로 구성되어 있습니다.

**· LSP**

: 14개의 Key Points가 있으며 스포츠 경기 중인 이미지를 수집하여 단일인물로 만든 데이터셋으로 1만장의 이미지로 구성되어 있습니다.

프로젝트의 주제가 걸음걸이 이기 때문에 걸음걸이를 인식할 수 있도록 이러한 데이터 셋들 중에서 발목과 발부분의 Key Points가 있는 것을 우선시하였고 데이터셋의 접근성 또한 가장 우수하며, Pose estimation의 성능도 높은 정확도(Mean mAP: 77%)를 나타낸 **COCO+foot** 데이터셋을 활용하기로 하였습니다.

**- 데이터셋 별 Key Points 비교**



**6. 스택**

AI : Keras opencv -> openpose(coco+foot model)

DB : My SQL

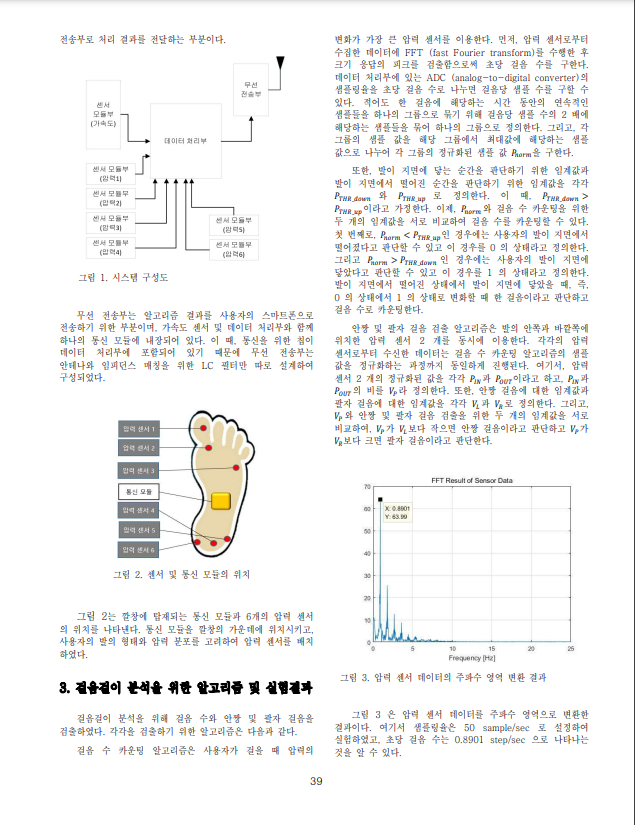
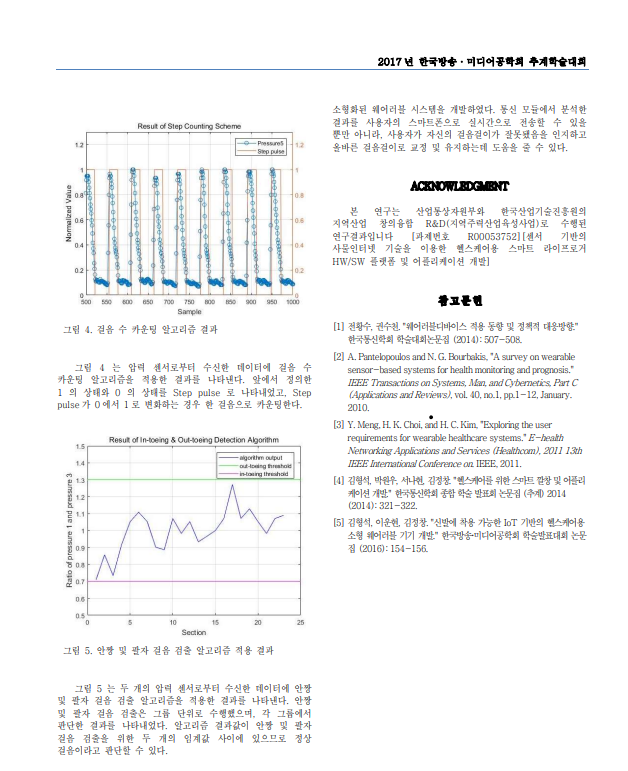
Front-End : React

Animation : Unity – 사용자가 3자 입장에서 객관적으로 비교 분석하기 위한 시각적인 도구

Back-End : Django

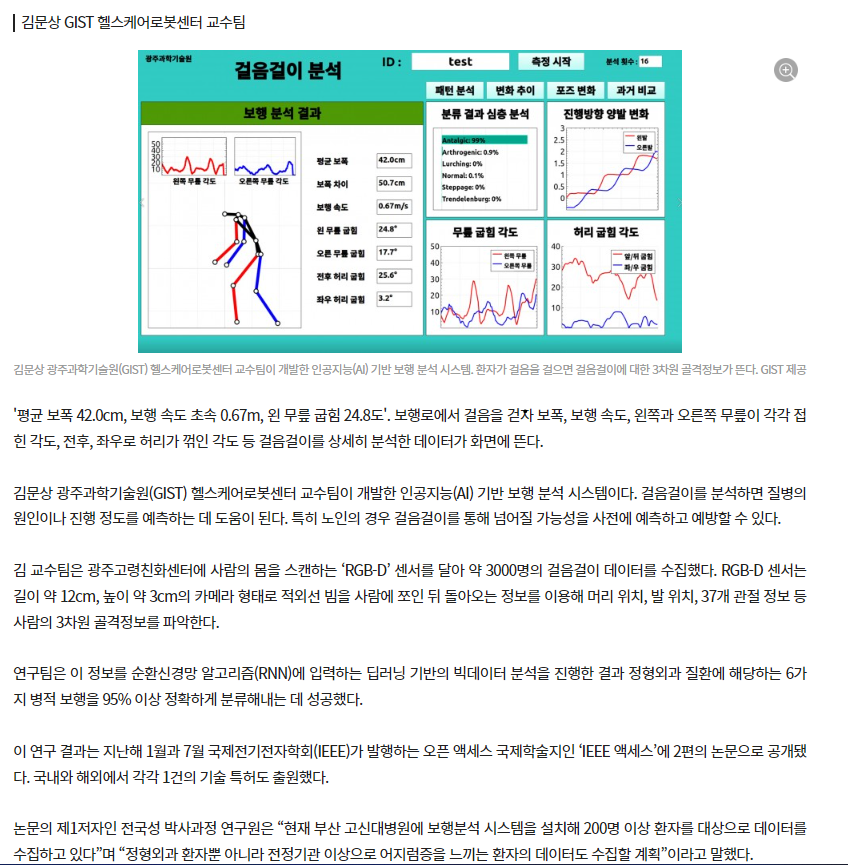
**7. 사례조사**

**-압력 센서를 이용한 걸음걸이 분석 사례-**

깔창에 탑재되는 통신 모듈과 6개의 압력 센서의 위치를 나타낸다. 통신 모듈을 깔창의 가운데에 위치시키고, 사용자의 발의 형태와 압력 분포를 고려하여 압력 센서를 배치하였다. 먼저, 압력 센서로부터 수집한 데이터에 FFT (fast Fourier transform)를 수행 후 초당 걸음 수를 구한다. 안짱 및 팔자 걸음 검출 알고리즘은 발의 안쪽과 바깥쪽에 위치한 압력 센서 2 개를 동시에 이용한다. 각각의 압력 센서로부터 수신한 데이터는 걸음 수 카운팅 알고리즘의 샘플 값을 정규화 하는 과정까지 동일하게 진행된다. 여기서, 압력 센서 2 개의 정규화 된 안짱 걸음에 대한 임계값과 팔자 걸음에 대한 임계값을 서로 비교하여 걸음의 종류를 구별할 수 있는 사례이다.

**- RGB-D센서를 이용한 걸음걸이 분석 사례-**

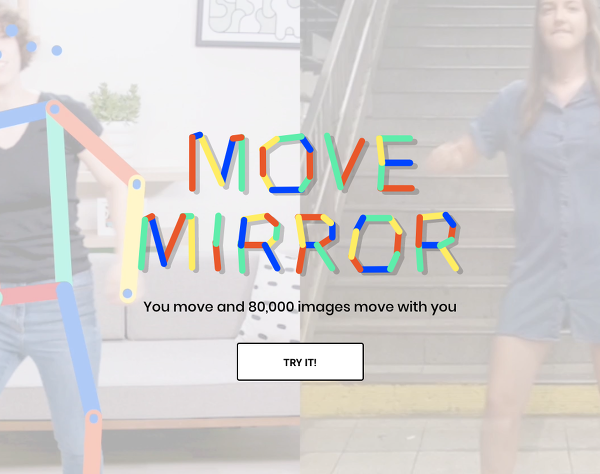


사람의 몸을 스캔하는 ‘RGB-D’ 센서를 달아 약 3000명의 걸음걸이 데이터를 수집했다. RGB-D 센서는 길이 약 12cm, 높이 약 3cm의 카메라 형태로 적외선 빔을 사람에 쪼인 뒤 돌아오는 정보를 이용해 머리 위치, 발 위치, 37개 관절 정보 등 사람의 3차원 골격정보를 파악합니다. 이 정보를 순환신경망 알고리즘(RNN)에 입력하는 딥러닝 기반의 빅데이터 분석을 진행한 사례이다.

\*한계점

걸음걸이 측정 및 구별을 위해 ‘RGB-D' 센서나 압력 센서와 같은 장치를 필요로 합니다.

**- PoseNet를 이용한 동작인식 사례(구글 무브미러)-**



웹켐이 장착되고 크롬 브라우저를 통해 인터넷에 연결된 노트북이나 PC를 이용하여 무브 미러 홈페이지에 접속한 후, 카메라 앞에서 서서 자유롭게 포즈를 취합니다. 그러면 웹켐으로 입력되는 영상을 모션 캡처 기능을 이용해 동작이나 자세를 추정합니다. 이를 구현하기 위해 구글은 17가지 신체 부위의 움직임을 감지할 수 있는 PoseNet모델과 Tensorflow.js를 이용하여 구현하였습니다. 동작을 추정할 때는 성별, 신장, 신체 유형 등의 개인적인 신체 특성은 고려되지 않으며 동작을 인식해 가장 유사한 동작의 사진을 검색할 수 있는 사례가 있습니다.

\*한계점

PoseNet은 OpenPose와 비교하여 직관적으로 동작을 인식하기 힘들고 스켈레톤이 깜빡거리고 사라지는 등 데이터 누락이 많습니다.



<https://www.youtube.com/watch?v=fPvvF1HsuJc>

**8. 스케줄**

· **4/1~4/25**

DB, React, Unity 구현 및 Django와 연동

- DB : 회원 관리에 필요한 데이터 베이스 구축

- React : 회원가입, 로그인 기능 구현

- Unity : 애니메이션 기본 움직임 설정, web내에서 시각화

**· 4/25** 설계 보고서 제출

**· 5/1~5/31**

- AI

: Open Pose를 활용한 사용자의 동영상과 올바른 걸음걸이 동영상의 스켈레톤 분석 및 비교 결과 데모 생성

- DB : 분석 결과에 따른 수치 입출력 확인

- React : 동영상 업로드 기능 구현 및 UI를 고려한 웹 사이트 구축

**· 5/23** 중간 보고서 제출

**· 6/6** 최종 보고서 및 구현 결과물 제출

**· 6/13** 최종 자료 제출