

YOLOv5 및 MediaPipe를 이용한
실시간 3대 운동 AI 자세 교정
서비스에 대한 연구

A study on Real-Time the big three exercises AI
posture correction service Using YOLOv5 and
MediaPipe

지도교수 박 소 현

동국대학교 과학기술대학 컴퓨터공학전공

고 영 민

2023. 11. 20

YOLOv5 및 MediaPipe를 이용한
실시간 3대 운동 AI 자세 교정
서비스에 대한 연구

A study on Real-Time the big three exercises AI
posture correction service Using YOLOv5 and
MediaPipe

고 영 민

지도교수 박 소 현

.

동국대학교 과학기술대학 컴퓨터공학전공

목 차

1. 서론	7
2. 이론적 배경	9
2.1 YOLOv5	9
2.2 MediaPipe	9
2.3 OpenCV	10
2.4 Streamlit	10
2.5 사이킷런(Scikit-Learn)	10
2.6 랜덤 포레스트 알고리즘(Random Forest Algorithm)	11
3. 제안하는 실시간 3대 운동 AI 자세 교정 서비스	12
3.1 시스템 아키텍처	12
3.2 운동을 수행 중인 사람 객체의 관절을 추정하는 알고리즘	12
3.3 운동 횟수 세기 알고리즘	13
3.4 운동별 자세 추론 및 교정 알고리즘	14
4. 실험	18
4.1 데이터 세트	18
4.2 실험 환경	23
4.3 성능 평가	23
5. 결론	28

참고문헌	29
------------	----

요약서	30
-----------	----

그 림 목 차

[그림 1] 종목별 생활체육 참여율 통계	8
[그림 2] 정적 이미지에 대한 피드백을 제공하는 관련 연구	8
[그림 3] YOLOv5 모델별 성능	9
[그림 4] MediaPipe Pose Landmark	10
[그림 5] 시스템 아키텍처	12
[그림 6] 운동을 수행 중인 사람 객체 탐지 후 관절 좌표 추정	13
[그림 7] 운동 횟수 세기 알고리즘	13
[그림 8] 관절 사이각 계산 알고리즘	14
[그림 9] 관절점을 이용하여 관절 사이각 계산하는 함수 의사코드	15
[그림 10] 벤치프레스 운동 수행 자세 분류 및 피드백 제공 예시	18
[그림 11] 스쿼트 운동 수행 자세 분류 및 피드백 제공 예시	19
[그림 12] 데드리프트 운동 수행 자세 분류 및 피드백 제공 예시	19
[그림 13] 벤치프레스 - 촬영 거치대 키퍼뷰	22
[그림 14] 벤치프레스 - 촬영 거치대 탑뷰	22
[그림 15] 스쿼트 및 데드리프트 - 촬영 거치대 키퍼뷰	23
[그림 16] 스쿼트 및 데드리프트 - 촬영 거치대 탑뷰	23
[그림 17] 벤치프레스 촬영 예시	24
[그림 18] 벤치프레스 운동 자세별 라벨링 개수	25
[그림 19] 스쿼트 운동 자세별 라벨링 개수	25
[그림 20] 데드리프트 운동 자세별 라벨링 개수	26
[그림 21] 좁은 여백의 라벨링 모습	27
[그림 22] 충분한 여백의 라벨링 모습	28
[그림 23] 운동을 수행 중인 사람 객체 탐지 후 관절 좌표 추정(벤치프레스)	28
[그림 24] 운동을 수행 중인 사람 객체 탐지 후 관절 좌표 추정(스쿼트)	29
[그림 25] 운동을 수행 중인 사람 객체 탐지 후 관절 좌표 추정(데드리프트)	29

표 목 차

[표 1] 주요 관절의 사이각	15
[표 2] 벤치프레스 자세 분류	16
[표 3] 스쿼트 자세 분류	16
[표 4] 데드리프트 자세 분류	17
[표 5] 올바른 자세에 대한 피드백	17
[표 6] 벤치프레스 수행 중 부적절한 자세에 대한 피드백	17
[표 7] 스쿼트 수행 중 부적절한 자세에 대한 피드백	17
[표 8] 데드리프트 수행 중 부적절한 자세에 대한 피드백	18
[표 9] 운동 종목별 카메라 위치	21
[표 10] 여백이 거의 없도록 훈련 데이터를 라벨링한 모델	27
[표 11] 충분한 여백을 포함하여 훈련 데이터를 라벨링한 모델	28
[표 12] 머신러닝 알고리즘별 벤치프레스 자세 분류 모델 성능 비교	30
[표 13] 랜덤 포레스트로 학습한 벤치프레스 자세 분류 모델의 성능 지표 결과	30
[표 14] 머신러닝 알고리즘별 스쿼트 자세 분류 모델 성능 비교	30
[표 15] 랜덤 포레스트로 학습한 스쿼트 자세 분류 모델의 성능 지표 결과	31
[표 16] 머신러닝 알고리즘별 데드리프트 자세 분류 모델 성능 비교	31
[표 17] 랜덤 포레스트로 학습한 데드리프트 자세 분류 모델의 성능 지표 결과	31

1. 서론

2022년 문화체육관광부의 국민 생활체육 조사의 결과[1]를 보면 조사대상 중 61.2%는 주 1회 이상 꾸준히 생활체육을 즐겨하고 있으며, 그중에서 보디빌딩은 12.8%로 걷기 다음인 2위로 상당히 높은 비율을 차지하고 있다(그림 1). 보디빌딩의 핵심 운동은 3대 운동인 '벤치프레스', '스쿼트', '데드리프트'가 있다. 하지만 3대 운동은 주로 고중량을 다루는 운동이므로 잘못된 자세로 운동을 수행하는 경우, 큰 부상을 초래할 위험이 크다. 그러나 대다수는 헬스장에 설치되어 있는 거울에만 의존하여 스스로 자세를 파악하며 운동을 수행한다. 하지만 전문적인 트레이너가 자세를 봐주면서 실시간으로 피드백을 하는 경우가 아닌 이상 스스로 자세의 옳고 그름을 판단하기 어렵다.

최근 부적절한 자세로 웨이트 트레이닝을 수행하여 부상을 초래하는 문제를 해결하기 위해 MeidaPipe나 OpenPose와 같은 자세 추정 프레임워크를 활용한 연구가 나타나고 있다. 이러한 문제를 해결하기 위한 방법에는 MediaPipe를 통해 골격 정보를 활용해 관절 점의 각도를 계산하여 기준 범위를 정해 옳고 그름을 판단하거나 단순히 운동 횟수만 안내하는 방법[2], MoveNet을 통해 원본군(사용자)의 골격 정보와 대조군(전문 트레이너 영상)의 골격 정보 얻고 딥러닝을 통해 원본군과 대조군을 비교하여 각 관절별로 옳고 그름을 판단하는 방법[3], 정적인 이미지 파일을 또는 미리 촬영된 비디오 파일을 업로드하여 자세를 판단하는 방법이 있다.[4][5] 그림 2는 정적인 이미지에 대해서 자세 피드백을 제공하는 이미지이다.[4] 이와 같은 방법들은 단순히 전문가와 사용자의 관절점 데이터들의 유사점을 비교하거나 골격 정보를 활용하여 각 관절의 각도를 구한 값만을 활용하여 자세의 옳고 그름을 판단하여 정확도가 낮다는 한계가 있으며 관절점 데이터만을 이용하여 학습할 경우 좌표가 달라지면 예측 결과가 크게 달라진다는 문제점이 있다. 또한 일부 연구에서는 정적 이미지나 촬영된 비디오 파일에 대해서만 피드백하는 방식인데, 운동을 수행하는 동안에는 실시간으로 자세에 대한 피드백할 수 없다는 한계가 있다. 그 외에도 실시간으로 피드백을 제공하는 서비스 역시 단순히 옳고 그름만 판단하여 운동을 수행할 때 정확히 어떤 자세가 문제인지 알려주지 못한다는 한계점이 있다. 그리고 운동의 동작이 연속적인 동작을 수행하는 운동의 자세를 단순히 시작점과 끝점에 대해서만 파악하고는 정확한 운동 자세를 피드백하기 어렵다는 기술적 어려움이 있다. 이를 극복하기 위한 유사한 연구로는 딥러닝 모델인 CNN과 LSTM을 결합하여 요가를 수행하는 비디오 프레임의 연속적인 동작을 모니터링하여 피드백을 제공하는 연구가 있다.[6]

또한 국내에 등록된 실시간 운동 자세 교정 서비스로는 'GymMate'와 같은 서비스가 있다.[7] 'GymMate'는 스마트폰을 고정하여 수행할 운동을 선택한 다음 자세의 옳고 그름을 판단한다. 그러나 이 서비스 운동 수행자의 자세를 파악하여 단순히 Good이나 Bad라는 문구만을 사용자에게 전달할 뿐이다. 하지만 운동을 수행하며 문구를 운동 수행자가 일일이 확인하는 일은 비현실적이며 음성으로 피드백이 제공되어야 실시간으로 자세 교정이 가능하다.

본 논문에서는 위에서 언급한 기존 연구의 한계를 극복하기 위해 MediaPipe를 통해 추출한 관절 좌표 데이터 중에서 각 운동별로 사용되는 주요 관절만을 사용할 뿐만 아니라 주요 관절간의 각도를 계산한 값을 후처리하여 지도학습을 진행함으로써 모델의 정확도를 높이는 것을 목표로 한다. 이를 위해 본 논문에서는 수집된 관절점 데이터를 활용하여 관절의 사이각을 구할 수 있는 알고리즘을 제안한다. 그리고 MediaPipe의 경우 단일 인물에 대해서만 관절점 검출이 가능하므로 본 논문에서는 YOLOv5를 이용하여 운동을 수행하는 사람 객체만 감지하는 모델을 제안한다.

본 논문에서는 첫째, MediaPipe의 스켈레톤 인식 기술을 사용하여 관절점을 추출하고 이를 기반으로 관절의 사이각을 구하여 데이터를 수집한다. 이때 여러 각도에서 데이터를 수집함으로써 다양한 각도에서 비디오 데이터를 입력받더라도 비슷한 예측 결과를 기대할 수 있다. 둘째, 수집된 데이터를 전처리하여 로지스틱 회귀, 릿지 분류, 랜덤 포레스트, 그레디언트 부스팅의 4가지 알고리즘을 이용하여 자세 분류 모델을 구성하고 성능을 비교하여 최적화된 모델의 가중치를 얻는다. 셋째, Streamlit 라이브러리를 통해 웹 환경에서 실시간 비디오 프레임 입력받아 실시간으로 운동 자세를 인식하여 분류하고 해당 자세에 대한 세부적인 피드백을 제공한다. 이때, 웹상에서는 실시간으로 운동 횟수와 관절별 각도도 확인할 수 있다. 또한 제공되는 피드백은 텍스트뿐만 아니라 음성합성 기술(TTS)을 이용하여 제작한 음성을 출력하도록 함으로써 더욱 생동감 있게 피드백을 제공할 수 있는 서비스를 구현하였다.

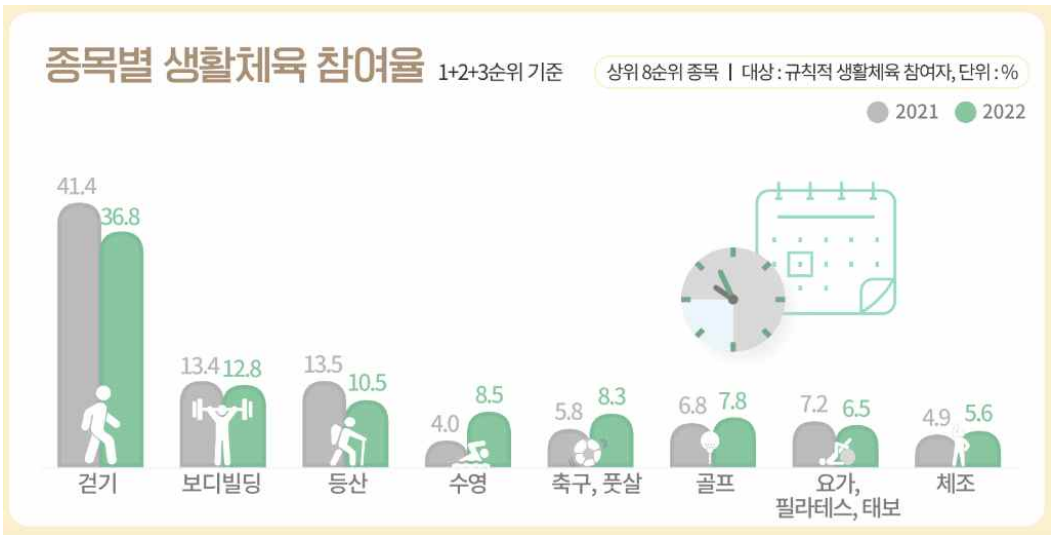


그림 1. 종목별 생활체육 참여율 통계

Image	Feedback
	Your knees are too caved in. Use a lighter weight. Your shoulders are uneven. Stretch them out before lifting and consider lowering the weight.
	Your knees are too caved in. Use a lighter weight.

그림 2. 정적 이미지에 대한 피드백을 제공하는 관련 연구

2. 이론적 배경

2.1 YOLOv5

YOLOv5(You Only Look Once version 5)는 컴퓨터 비전 및 객체 감지 분야에서 중요한 역할을 하는 딥러닝 기반의 객체 감지 알고리즘이다. YOLOv5는 YOLO 시리즈 중 비교적 최근에 제안된 5번째 버전으로, 빠른 속도와 높은 정확도를 제공하여 다양한 응용 분야에서 사용되고 있다. 이 알고리즘의 주요 특징 중 하나는 실시간 객체 감지를 가능하게 하는 높은 처리 속도로 인해 실시간 비디오 스트림에서 실시간 객체 검출을 수행하는 데 매우 유용하다. 그림 3은 YOLOv5의 모델별 성능을 나타낸 것이다.[8] 본 논문에서 관절점을 추출하기 위해 사용하는 MediaPipe의 경우 단일 인물에 대해서만 자세 추정을 할 수 있는데 운동을 수행하는 상황에서는 다른 인물이 카메라에 촬영될 수 있어 정확한 피드백을 제공하지 못할 수 있다. 이러한 점을 극복하기 위해 YOLOv5를 통해 운동을 수행 중인 객체를 탐지하여 해당 객체에 대해서만 자세 추정을 하도록 하는 운동을 수행 중인 사람 객체의 관절을 추정하는 알고리즘을 구현하였다. 그리고 실시간 영상처리를 위하여 연산 과정이 간단하고 가벼운 모델인 YOLOv5

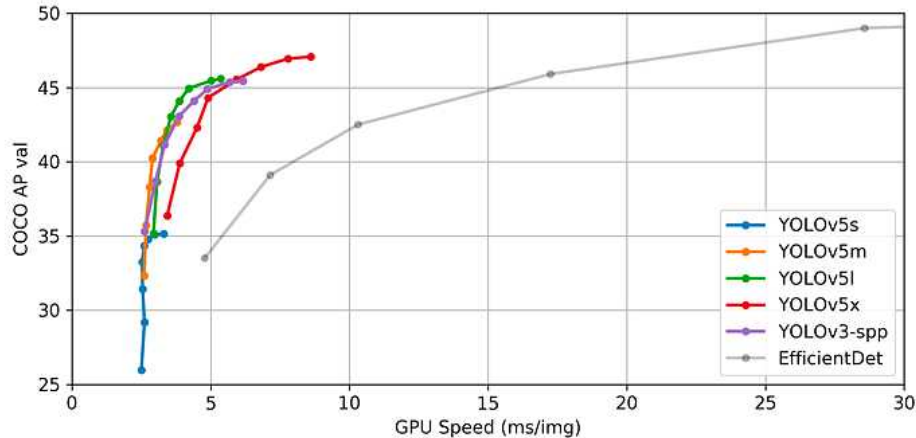


그림 3. YOLOv5 모델별 성능

2.2 미디어파이프(MediaPipe)

미디어파이프(MediaPipe)는 구글(Google)에서 개발한 오픈소스 프레임워크로, 비전 및 머신러닝을 기반으로 한 컴퓨터 비전 응용 프로그램을 개발하는 데 사용되며 특히 실시간 데이터 스트림에서 작동하도록 설계되어 있다. 미디어파이프는 객체 감지, 자세 추정, 얼굴 감지, 손 추적 등과 같은 다양한 비전 작업을 지원한다.[9] 그중에서 자세 추정(Pose Estimation)을 활용하면 그림 4와 같이 33개의 관절 좌표 데이터를 추출할 수 있다.

본 논문에서는 모든 관절의 좌표를 사용하지 않고 운동 종목별로 주요 관절의 좌표만을 사용한다. 모든 관절의 좌표를 사용하지 않은 이유는 얼굴 포인트(1, 2, 3, 4, 5, 6, 7, 8, 9, 10)와 손가락을 나타내는 포인트(17, 18, 19, 20, 21, 22)는 운동 자세에 큰 영향을 끼치지 않으며 오히려 학습에 방해가 될 수 있기 때문이다.

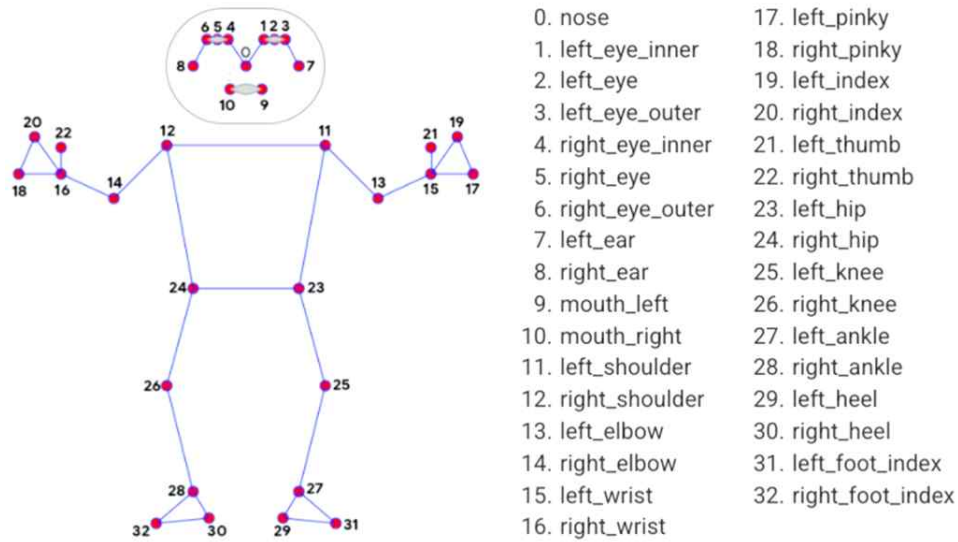


그림 4. MediaPipe Pose Landmark

2.3 OpenCV

OpenCV(Open Source Computer Vision Library)는 컴퓨터 비전 및 이미지 처리 작업을 위한 오픈소스 라이브러리로 다양한 프로그래밍 언어에서 사용할 수 있다. OpenCV는 이미지 및 비디오 처리, 객체 감지, 얼굴 인식, 자동차 번호판 인식, 캘리브레이션 등 다양한 컴퓨터 비전 작업을 지원하며, 컴퓨터 비전 연구 및 응용 프로그램 개발에 널리 사용된다.

본 논문에서는 파이썬 프로그래밍 언어를 통해 OpenCV 라이브러리를 사용한다.

2.4 Streamlit

Streamlit은 데이터 과학 및 웹 애플리케이션 개발을 위한 파이썬 기반의 오픈소스 라이브러리이다. Streamlit을 사용하면 데이터 시각화, 웹 애플리케이션, 대시보드 등을 빠르고 간편하게 개발할 수 있으며, 데이터 과학자 및 엔지니어들이 데이터를 시각화하고 공유하는 데 매우 유용한 도구이다.

본 논문에서는 Streamlit을 통해 실시간으로 이미지 처리를 수행함으로써, 운동 자세에 대한 피드백을 사용자에게 실시간으로 제공한다.

2.5 사이킷런(Scikit Learn)

사이킷런(Scikit Learn)은 파이썬 기반의 머신러닝 라이브러리로, 데이터 과학 및 기계 학습 모델을 빠르고 효율적으로 구현하고 적용하는 데 사용되는 강력한 도구이다. 이 라이브러리는 다양한 머신러닝 알고리즘과 도구를 제공하여 데이터 분석, 패턴 인식, 예측 및 분류와 같은 다양한 머신러닝 작업을 지원한다.

본 논문에서는 각 운동별 옳은 자세와 부적절한 자세의 비디오 데이터를 수집하고 관절점(joint landmarks) 데이터를 수집하여 각 자세별로 지도학습이 가능하도록 라벨링을 진행한다. 이 데이터를 이용하여 로지스틱 회귀, 릿지 분류, 랜덤 포레스트, 그래디언트 부스팅 알고리즘으로 모델을 구현하기 위해 사이킷런 라이브러리를 사용한다.

2.6 랜덤 포레스트 알고리즘(Random Forest Algorithm)

랜덤 포레스트 알고리즘(Random Forest Algorithm)은 머신러닝 분야에서 널리 사용되며 성능이 높은 앙상블(Ensemble) 학습 방법 중 하나이다. 이 알고리즘은 주로 분류(Classification)와 회귀(Regression) 작업에 사용되며, 데이터 과학 및 기계 학습 분야에서 높은 예측 성능과 안정성으로 인해 많이 사용된다.

본 논문에서는 벤치프레스, 스쿼트, 데드리프트 각각의 운동 자세 분류 모델에 대해 여러 알고리즘별로 성능 지표를 비교한 결과 랜덤 포레스트 알고리즘 이용하여 학습한 모델에서 F1-Score가 가장 높았으므로 본 논문에서 해당 알고리즘을 사용한다.

3. 제안하는 실시간 3대 운동 AI 자세 교정 서비스

3.1 시스템 아키텍처

본 논문에서 제안하는 시스템 구성도는 그림 5와 같다. 제안하는 실시간 3대 운동 AI 자세 교정 서비스는 Streamlit을 이용하여 서버를 구축한다. 먼저, 사용자의 모바일 카메라로 촬영된 모든 프레임을 OpenCV를 활용해 실시간 영상으로 처리하여 서버에 전송한다(그림 5, 1. input). 그리고 서버에서는 전달받은 영상의 모든 프레임에 대해서 YOLOv5를 이용해 구현한 운동을 수행 중인 사람 객체를 탐지하는 모델을 통해 바운딩 박스가 가장 큰 객체를 실제 운동 중인 객체로 판단하고, 가장 큰 바운딩 박스 내부에 있는 객체에 대해서만 MediaPipe 프레임워크를 사용해 운동을 수행하는 사람의 관절 좌표 정보를 추정하고 각 운동별로 머신러닝 알고리즘을 통해 학습한 모델을 이용하여 자세 분류 결과를 서버에게 반환한다(그림 5, 2. frame & 3. classification). 마지막으로 서버에서는 반환된 결과를 분석해 운동 횟수를 카운트하고 잘못된 자세에 대해서는 화면에 부적절한 자세에 대한 피드백을 화면과 음성으로 제공한다(그림 5, 4. feedback).

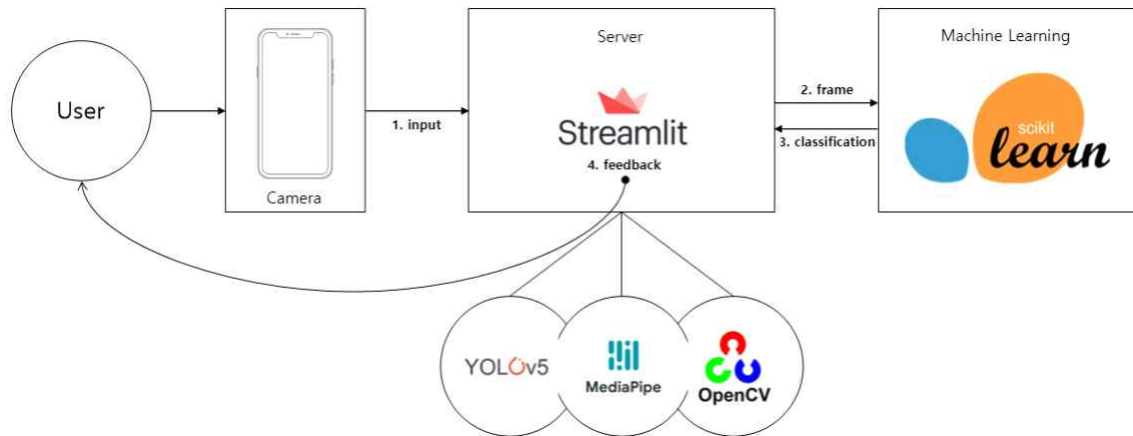


그림 5. 시스템 아키텍처

3.2 운동을 수행 중인 사람 객체의 관절을 추정하는 알고리즘

MediaPipe는 모바일에서도 실시간 추론을 할 수 있는 매우 빠른 모델이지만 여러 사람이 아닌 오직 한 사람만 추적하여 관절 좌표를 추정할 수 있다. 하지만 헬스장과 같이 여러 사람이 카메라 화면에 동시에 촬영될 수 있는 환경에서는 자세 추정의 정확도가 낮아질 수 있으며 정확하지 않은 피드백을 제공할 위험이 있다. 따라서 본 논문에서는 이러한 한계를 극복하기 위하여 그림 6에 제시한 운동을 수행 중인 사람 객체의 관절을 추정하는 알고리즘을 제안한다.

본 논문에서 제안하는 운동을 수행 중인 사람 객체의 관절을 추정하는 알고리즘은 사람과 관련된 데이터 세트를 YOLOv5 모델로 학습시킨 후 그 가중치를 사용하여 사용자의 웹캠으로부터 입력된 영상 데이터 중 사람 객체를 탐지한다. 그리고 가장 큰 바운딩 박스(bbox)를 가진 객체를 실제 운동 중인 사람 객체라고 판단하고 해당 바운딩 박스 내부에 있는 객체에 대해서 MediaPipe 프레임워크를 활용해 관절 좌표를 추정한다.

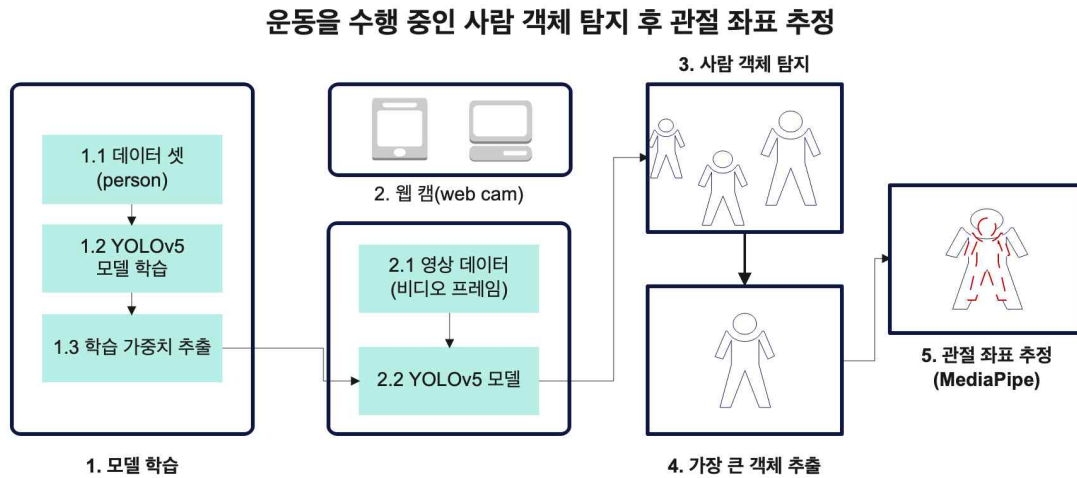


그림 6. 운동을 수행 중인 사람 객체 탐지 후 관절 좌표 추정

3.3 운동 횟수 세기 알고리즘

본 논문에서 구현한 운동 횟수 세기 알고리즘은 그림 7과 같다. counter 변수에 0을 초기화하고 현재 상태를 나타내는 current_stage 변수를 공백으로 초기화한다. 그리고 무한 루프를 돌면서 서버에 학습된 모델의 예측값과 비교한다. 모델 예측값에 'down'이 포함되어 있으면 현재 상태를 'down'으로 설정하고, 그렇지 않으면 현재 상태를 'up'으로 설정하고 counter 변수를 1 증가시킨다.

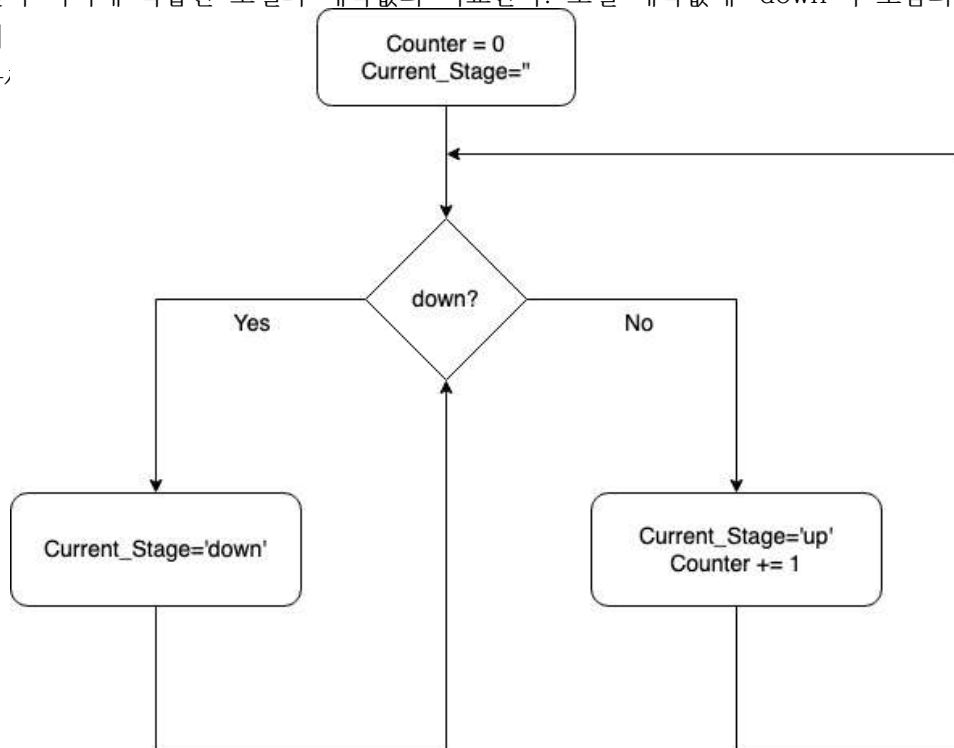


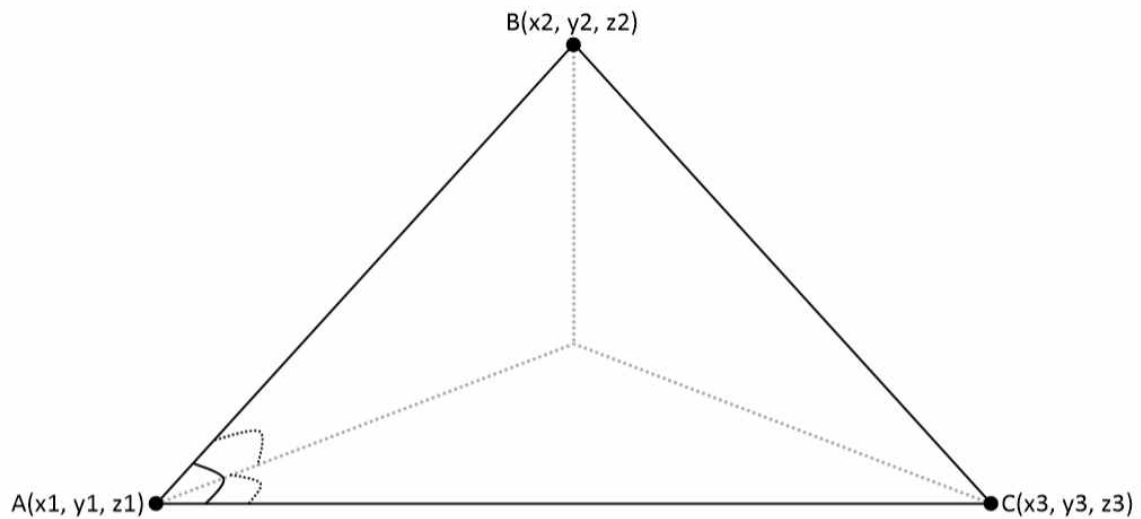
그림 7. 운동 횟수 세기 알고리즘 순서도

3.4 운동별 자세 추론 및 교정 알고리즘

3.4.1 관절 사이각 계산 알고리즘

3.2절에서 추출된 관절 좌표는 각 관절점(joint landmarks)에 대한 위치 정보($l = (x, y, z)$)만을 담고 있다. 관절점에 대한 위치 정보 자체는 무의미한 값이지만 이러한 정보를 통해 관절의 사이각을 구하면 운동 자세 분석에서 유의미한 정보로 사용할 수 있다.

삼각함수를 활용하면 관절점의 위치 정보를 통해 관절 간의 사이각(joint angle)을 구할 수 있는데 그림 8은 관절 사이각의 위치와 수식이다. 연산 방법은 $x1, y1$ 과 $x1, z1$ 또는 $y1, z1$ 을 기준으로 세 점이 이루는 사이각과 $x1, y1$ 을 기준 원점 축에 대한 각도 2개를 구한 뒤 그 값을 빼는 방법이다.



$$\theta = \left(\arctan \frac{y2 - y1}{x2 - x1} - \arctan \frac{y3 - y1}{x3 - x1} \right) * \left| \frac{180}{\pi} \right|$$

그림 8. 관절 사이각 계산 알고리즘

위의 알고리즘을 통해 왼쪽 팔꿈치의 각도를 구할 때의 의사코드는 그림 9와 같다. 변수 a, b, c에 각각 왼쪽 어깨, 왼쪽 팔꿈치, 왼쪽 손목 관절점을 대입하여 np.arctan2() 함수를 활용해 이들 간의 각도를 구하면 라디안으로 나타나는데 $180/\text{np.pi}$ 를 곱해 라디안을 다시 도(°) 단위로 변환한다. 만약 각도가 180° 가 넘으면 360° 에서 구해진 각도를 뺀 값을 반환한다.

```

import math
def calculateAngle(a, b, c):
    a = np.array(a) # first
    b = np.array(b) # mid
    c = np.array(c) # end

    radians = np.arctan2(c[1]-b[1], c[0]-b[0]) - np.arctan2(a[1]-b[1], a[0]-b[0])
    angle = np.abs(radians*180.0 / np.pi)

    if angle > 180.0:
        angle = 360 - angle

    return angle

```

그림 9. 관절점을 이용하여 관절 사이각 계산하는 함수 의사코드

다음의 표 1은 각 운동별로 사용되는 관절점과 주요 관절의 사이각을 나타낸다. 목 관절의 사이각의 경우 왼쪽 어깨, 코, 왼쪽 골반의 사이각과 오른쪽 어깨, 코, 오른쪽 골반의 사이각의 산술 평균을 구하고 어깨, 팔꿈치, 골반, 무릎, 발목 관절의 사이각은 왼쪽 관절과 오른쪽 관절의 사이각을 각각 구한다.

Angle	사용한 관절점(joint landmark) 번호
목((좌+우)/2)	좌: 11, 0, 23(left shoulder, nose, left hip) 우: 12, 0, 24(right shoulder, nose, right hip)
어깨(좌, 우)	좌: 13, 11, 23(left elbow, left shoulder, left hip) 우: 14, 12, 24(right elbow, right shoulder, right hip)
팔꿈치(좌, 우)	좌: 11, 13, 15(left shoulder, left elbow, left wrist) 우: 12, 14, 16(right shoulder, right elbow, right wrist)
골반(좌, 우)	좌: 11, 23, 25(left shoulder, left hip, left knee) 우: 12, 24, 26(right shoulder, right hip, right knee)
무릎(좌, 우)	좌: 23, 25, 27(left hip, left knee, left ankle) 우: 24, 26, 28(right hip, right knee, right ankle)
발목(좌, 우)	좌: 25, 27, 29(left knee, left ankle, left heel) 우: 26, 28, 30(right knee, right ankle, right heel)

표 1. 주요 관절의 사이각

3.4.2 벤치프레스

다음 표 2와 같이 벤치프레스를 수행하는 자세별로 라벨링이 된 관절점 훈련 데이터 세트를 scikit-learn 라이브러리의 pipeline 클래스를 이용하여 로지스틱 회귀(Logistic Regression), 릿지 분류(Ridge Classifier), 랜덤 포레스트 분류(Random Forest Classifier), 그레이디언트 부스팅 분류(Gradient Boosting Classifier) 모델을 하나의 과정으로 훈련하도록 구현하였다.

본 논문에서는 각 모델별 성능을 평가하여 F1-Score가 가장 높은 랜덤 포레스트 모델을 선택하였다.

상태	자세 분류	단축성 수축(up)	신장성 수축(down)
Correct	올바른 자세	b_correct_up	b_correct_down
Incorrect	허리가 과도한 아치 자세	b_excessive_arch_up	b_excessive_arch_down
	바를 너무 넓게 잡은 자세	b_arms_spread_up	b_arms_spread_down

표 2. 벤치프레스 자세 분류

3.4.3 스쿼트

다음 표 3과 같이 스쿼트를 수행하는 자세별로 라벨링이 된 관절점 훈련 데이터 세트를 scikit-learn 라이브러리의 pipeline 클래스를 이용하여 로지스틱 회귀(Logistic Regression), 릿지 분류(Ridge Classifier), 랜덤 포레스트 분류(Random Forest Classifier), 그레이디언트 부스팅 분류(Gradient Boosting Classifier) 모델을 하나의 과정으로 훈련하도록 구현하였다.

본 논문에서는 각 모델별 성능을 평가하여 F1-Score가 가장 높은 랜덤 포레스트 모델을 선택하였다.

상태	자세 분류	단축성 수축(up)	신장성 수축(down)
Correct	올바른 자세	s_correct_up	s_correct_down
Incorrect	척추 중립이 아닌 자세	s_spine_neutral_up	s_spine_neutral_down
	무릎이 움푹 들어간 자세	s_caved_in_knees_up	s_caved_in_knees_down
	발을 너무 넓게 벌린 자세	s_feet_spread_up	s_feet_spread_down

표 3. 스쿼트 자세 분류

3.4.4 데드리프트

다음 표 4와 같이 데드리프트를 수행하는 자세별로 라벨링이 된 관절점 훈련 데이터 세트를 scikit-learn 라이브러리의 pipeline 클래스를 이용하여 로지스틱 회귀(Logistic Regression), 릿지 분류(Ridge Classifier), 랜덤 포레스트 분류(Random Forest Classifier), 그레이디언트 부스팅 분류(Gradient Boosting Classifier) 모델을 하나의 과정으로 훈련하도록 구현하였다.

본 논문에서는 각 모델별 성능을 평가하여 F1-Score가 가장 높은 랜덤 포레스트 모델을 선택하였다.

상태	자세 분류	신장성 수축(up)	단축성 수축(down)
Correct	올바른 자세	d_correct_up	d_correct_down
Incorrect	척추 중립이 아닌 자세	d_spine_neutral_up	d_spine_neutral_down
	바를 너무 넓게 잡은 자세	d_arms_spread_up	d_arms_spread_down
	바를 너무 좁게 잡은 자세	d_arms_narrow_up	d_arms_narrow_down

표 4. 데드리프트 자세 분류

다음 표5, 표6, 표7, 표8은 올바른 자세와 각 운동별 부적절한 자세에 대한 피드백에 대한 내용이다.

벤치프레스, 스쿼트, 데드리프트
올바른 자세 올바른 자세로 운동을 수행하고 있습니다.

표 5. 올바른 자세에 대한 피드백

벤치프레스
허리가 과도하게 아치 자세 허리를 너무 아치 모양으로 만들지 말고, 자연스럽게 피하도록 노력하세요.
허리가 과도하게 아치 자세 골반을 조금 더 들어올리고 복부를 긴장시켜 허리를 평평하게 유지하세요.
바를 너무 넓게 잡은 자세 바를 너무 넓게 잡았습니다. 조금 더 좁게 잡으세요.
바를 너무 넓게 잡은 자세 바를 잡을 때 어깨 너비보다 약간만 넓게 잡는 것이 좋습니다.

표 6. 벤치프레스 수행 중 부적절한 자세에 대한 피드백

스쿼트
척추 중립이 아닌 자세 척추가 과도하게 굽지 않도록 노력하세요.
척추 중립이 아닌 자세 가슴을 들어올리고 어깨를 뒤로 넣으세요.
무릎이 움푹 들어간 자세 스쿼트를 할 때 무릎이 움푹 들어가지 않도록 주의하세요.
무릎이 움푹 들어간 자세 엉덩이를 뒤로 빼서 무릎과 발끝을 일직선으로 유지하세요.
발을 너무 넓게 벌린 자세 발을 좁혀 어깨 너비 정도로만 벌리세요.

표 7. 스쿼트 수행 중 부적절한 자세에 대한 피드백

데드리프트
척추 중립이 아닌 자세 척추가 과도하게 굽지 않도록 노력하세요.
척추 중립이 아닌 자세 가슴을 들어올리고 어깨를 뒤로 넣으세요.
바를 너무 넓게 잡은 자세 바를 너무 넓게 잡았습니다. 조금 더 좁게 잡으세요.
바를 너무 넓게 잡은 자세 바를 잡을 때 어깨 너비보다 약간만 넓게 잡는 것이 좋습니다.
바를 너무 좁게 잡은 자세 바를 어깨 너비보다 조금 넓게 잡는 것이 좋습니다.

표 8. 데드리프트 수행 중 부적절한 자세에 대한 피드백

다음 그림 10, 그림 11, 그림 12은 실제 본 서비스가 벤치프레스와 스쿼트 그리고 데드리프트를 수행하는 객체의 관절점을 실시간으로 모니터링하며 수행 자세에 대해 피드백을 제공하는 모습이다.



그림 10. 벤치프레스 운동 수행 자세 분류 및 피드백 제공 예시

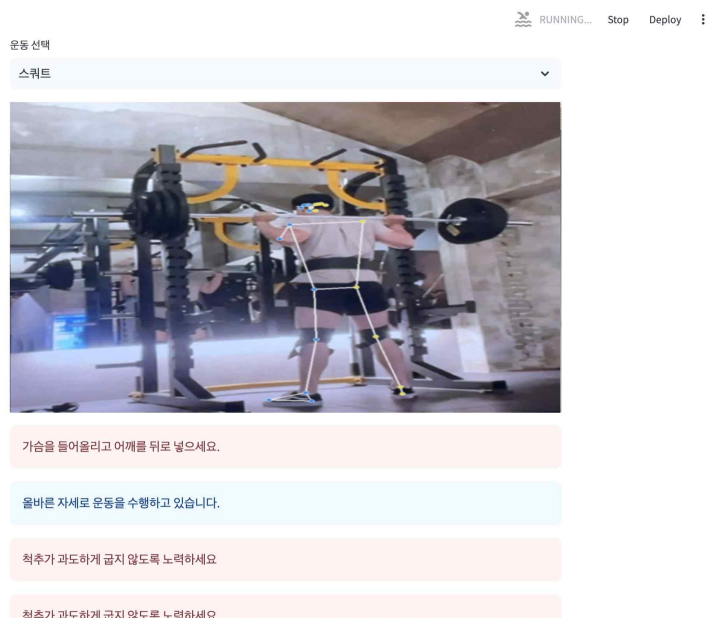
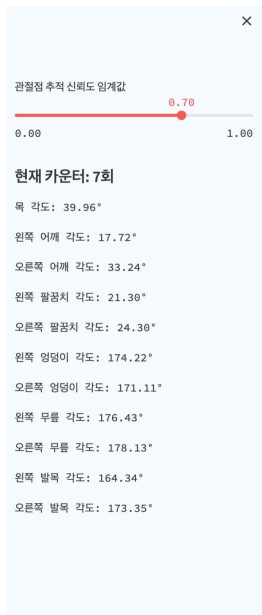


그림 11. 스쿼트 운동 수행 자세 분류 및 피드백 제공 예시

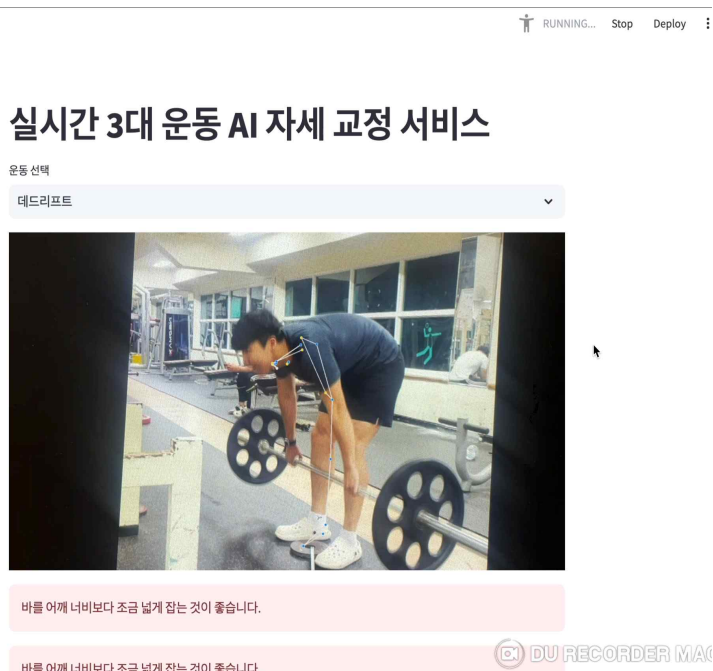
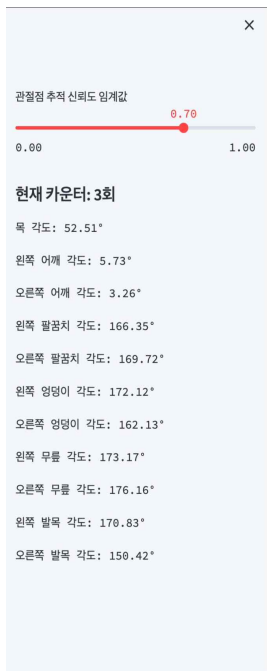


그림 12. 데드리프트 운동 수행 자세 분류 및 피드백 제공 예시

4. 실험

4.1은 본 논문에서 제안한 운동을 수행 중인 사람 객체 탐지 모델을 구현하기 위해 사용한 데이터 세트에 관한 설명과 운동 자세를 분류하는 모델을 구현하기 위해 직접 수집한 데이터 세트에 관한 설명이다. 다음으로 4.2에서는 본 논문의 실험에 사용된 실험 환경을 소개한다. 마지막으로 4.3에서는 구현한 각각의 모델에 대한 성능 평가가 이루어진다.

4.1 데이터 세트

4.1.1 운동을 수행 중인 사람 객체 탐지 모델 데이터 세트

본 논문에서는 운동을 수행 중인 사람 객체를 탐지하는 모델 구축을 위해 Roboflow에서 제공하는 데이터 세트를 사용하였고 일부 부족한 데이터에 대해서는 구글 이미지 데이터를 크롤링하여 추가로 수집하였다. 이후 Kaggle에서 제공하는 다양한 자세별 이미지 데이터 세트를 취합하여 사용하였다.

본 논문은 운동과 관련된 자세를 활용하므로 벤치프레스, 스쿼트, 데드리프트를 수행하는 이미지 데이터 세트를 수집하였고 운동 이미지만으로 학습했을 때 발생하는 과대적합 문제를 예방하기 위해 일반적인 사람 이미지, 공사장 인부 이미지, 누워있는 사람 이미지 등 Roboflow와 구글 이미지로부터 총 805장의 이미지를 수집하였으며 이후 Kaggle의 'Silhouettes of human posture' 데이터 세트에서 앞으로 기울인 자세, 누워있는 자세, 앉아 있는 자세, 서 있는 자세 각 1,200장씩 4,800장을 추가로 수집하여 총 5,605장의 이미지를 모델 학습에 사용하였다.

본 논문에서는 YOLOv5를 이용하여 운동을 수행 중인 사람 객체 탐지 모델을 구현하였는데 이를 위해 이미지 라벨링 툴인 Dark Label을 이용하였다.

본 논문의 실험에서는 전처리 작업을 총 두 번에 걸쳐 진행하였는데, 첫 번째 모델 학습을 위해 데이터를 전처리했을 때는 사람의 머리끝부터 발끝까지 최대한 여백이 없도록 라벨링 작업을 진행했고 사람 객체를 탐지하는 성능은 mAP_{0.5:0.95} 기준으로 0.843의 성능을 보였다. 하지만 해당 모델에서 검출한 객체를 기준으로 MediaPipe의 자세 추정을 진행하였을 때 부정확한 관절점을 추출하는 문제가 발생했다. 이후 데이터 트를 다시 라벨링 작업을 하였고 이전과는 다르게 사람 객체를 기준으로 상하좌우 모두 여백을 충분히 주어 다시 학습을 진행했고 성능은 mAP_{0.5:0.95} 기준으로 0.705의 성능을 보여 이전보다 0.138이 낮아졌지만, MediaPipe의 자세 추정과 결합하였을 때는 더 우수한 성능을 보였다.

4.1.2 운동 자세 분류 데이터 세트

본 논문에서는 벤치프레스, 스쿼트, 데드리프트 각 운동별 자세 분류 모델을 구현하기 위해 직접 데이터 세트를 수집하였다. 사람의 관절점(joint landmarks)을 추론할 때 카메라의 위치와 촬영 방향은 중요한 역할을 한다. 따라서 높은 정확도로 관절점을 추정하기 위해 표 9와 같이 모든 운동 종목에 대해서 운동을 수행 중인 대상 객체를 기준으로 정면에서 200cm, 좌우 대각선 방향으로 180cm 떨어진 위치에서 촬영하였다.

운동 종목	정면(cm)	대각선(cm)
벤치프레스	200	180
스쿼트	200	180
데드리프트	200	180

표 9. 운동 종목별 카메라 위치

그림 13은 벤치프레스 데이터 세트를 수집하기 위해 촬영 거치대를 정면과 좌우 대각선에 설치한 모습을 쿼터뷰로 나타낸 그림이다.

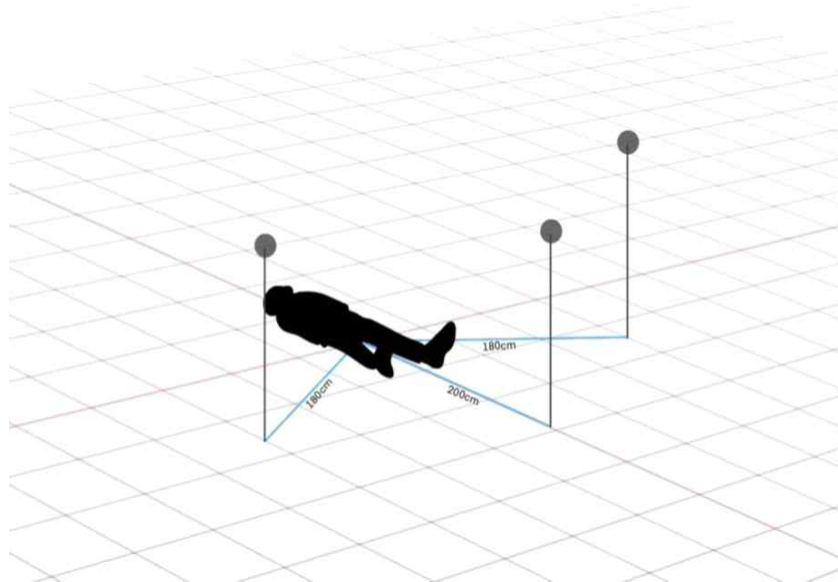


그림 13. 벤치프레스 - 촬영 거치대 쿼터뷰

그림 14는 벤치프레스 데이터 세트를 수집하기 위해 촬영 거치대를 정면과 좌우 대각선에 설치한 모습을 탑뷰로 나타낸 그림이다.

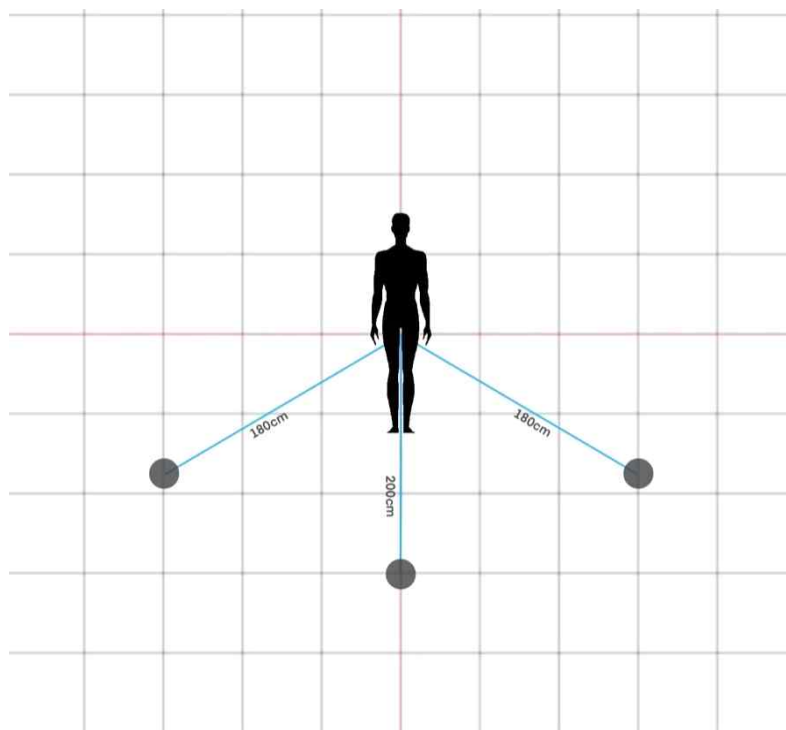


그림 14. 벤치프레스 - 촬영 거치대 위치 탑뷰

그림 15는 스쿼트와 데드리프트의 데이터 세트를 수집하기 위해 촬영 거치대를 정면과 좌우 대각선에 설치한 모습을 쿼터뷰로 나타낸 그림이다.

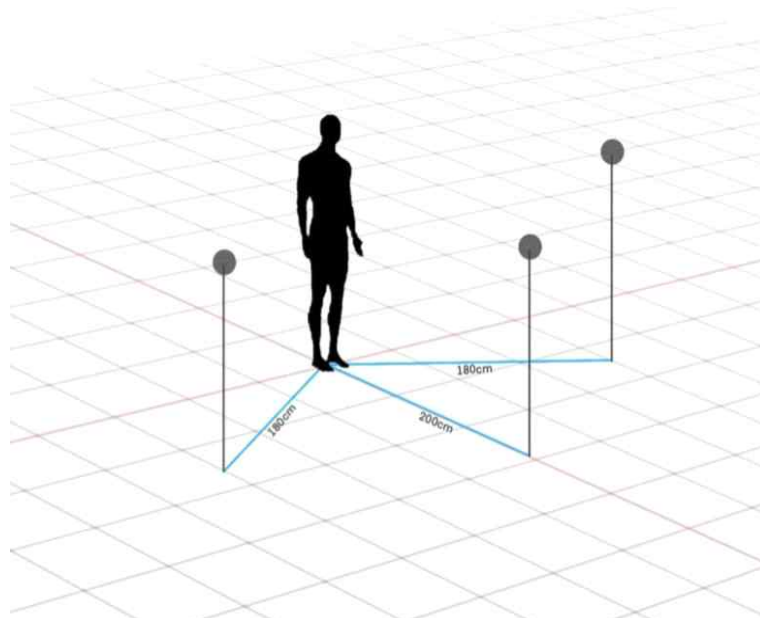


그림 15. 스쿼트 및 데드리프트 - 촬영 거치대 쿼터뷰

그림 16은 스쿼트와 데드리프트의 데이터 세트를 수집하기 위해 촬영 거치대를 정면과 좌우 대각선에 설치한 모습을 탑뷰로 나타낸 그림이다.

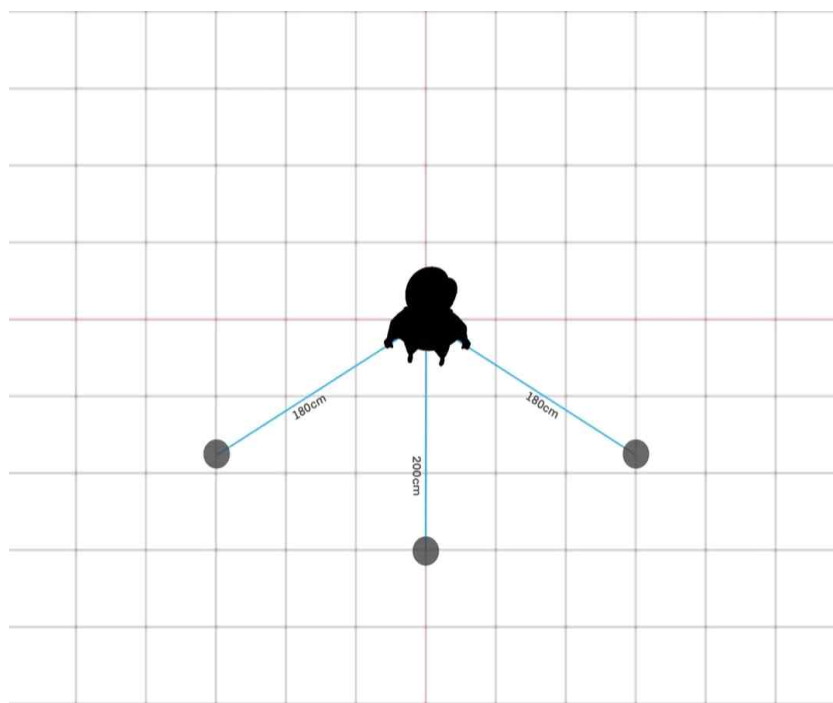


그림 16. 스쿼트 및 데드리프트 - 촬영 거치대 탑뷰

그림 17은 벤치프레스의 데이터 세트를 수집하기 위해 촬영 거치대를 좌측 대각선 방향, 정면, 우측 대각선 방향에 설치한 모습을 각각 나타낸 그림이다.

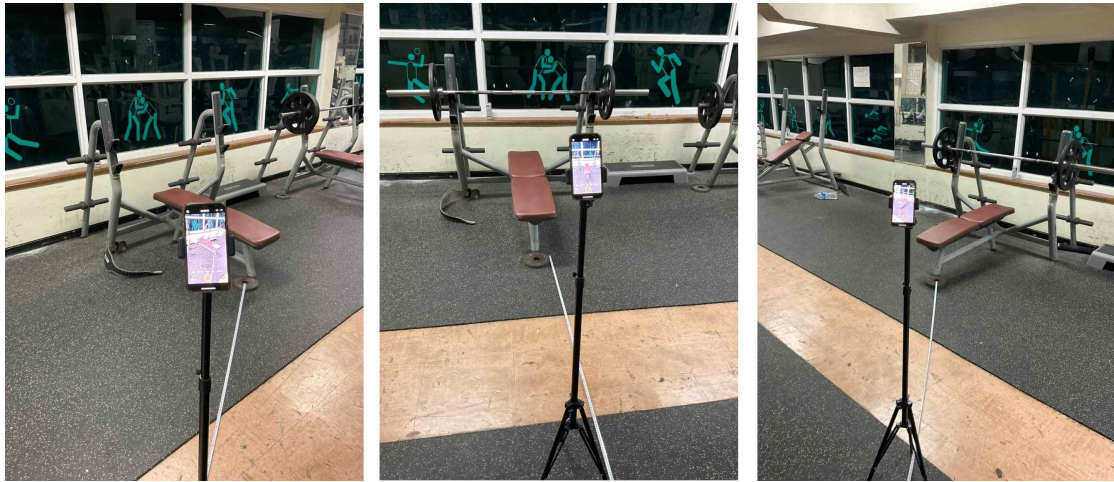
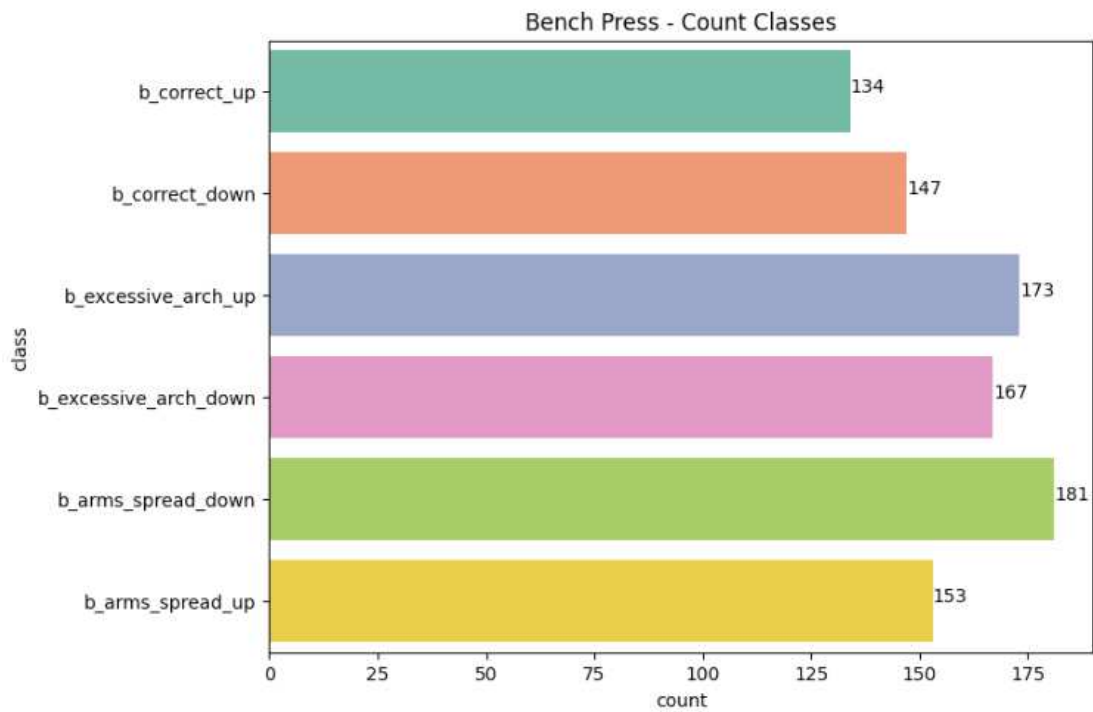


그림 17. 벤치프레스 촬영 예시

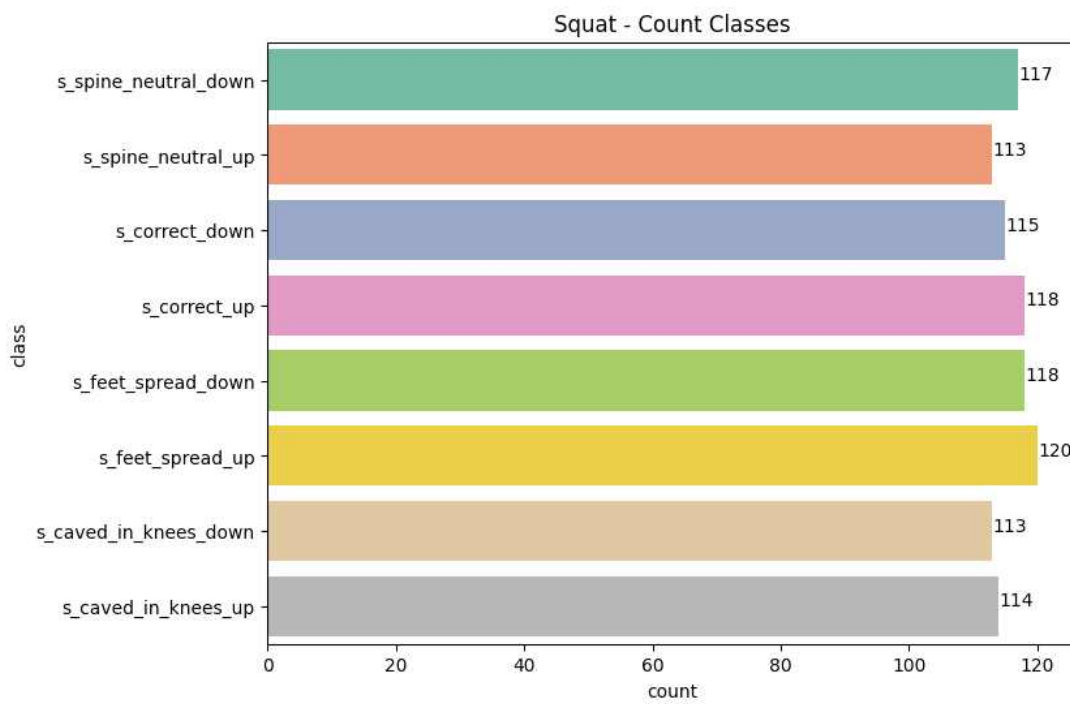
벤치프레스는 자세별로 총 10회의 운동을 반복 수행하여 총 12개의 비디오를 수집하였고 스쿼트와 데드리프트는 자세별로 총 10회의 운동을 반복 수행하여 각각 16개의 비디오를 수집하였다. 최종적으로 벤치프레스와 스쿼트, 데드리프트에 대해 각 카메라 위치별 44개의 비디오 데이터를 수집하였다.

수집한 비디오 데이터를 OpenCV 라이브러리와 MediaPipe 라이브러리를 활용하여 비디오 데이터의 각 프레임별로 수작업을 통해 라벨링을 진행하였고 각 운동별 라벨링된 개수를 나타낸 것이 그림 18, 그림 19, 그림 20이다. 벤치프레스는 6개의 클래스에 대해 955개의 라벨링한 데이터를 수집했고, 스쿼트는 8개의 클래스에 대해 928개의 라벨링한 데이터를 수집했으며, 마지막으로 데드리프트는 8개의 클래스에 대해 1,140개의 라벨링한 데이터를 수집했다.



counts: 955

그림 18. 벤치프레스 운동 자세별 라벨링 개수



counts: 928

그림 19. 스쿼트 운동 자세별 라벨링 개수

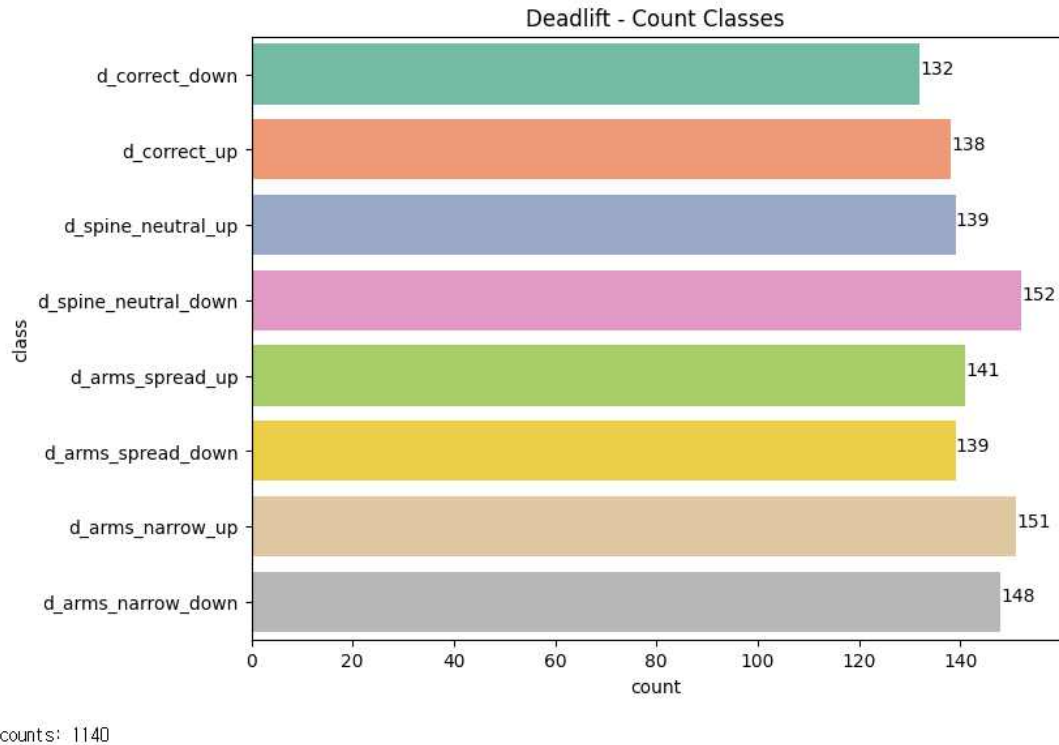


그림 20. 데드리프트 운동 자세별 라벨링 개수

벤치프레스, 스쿼트, 데드리프트 각각의 데이터 세트를 scikit-learn 라이브러리의 model_selection 모듈을 이용하여 모델의 훈련 데이터(training data)와 테스트 데이터(test data)를 7:3 비율로 나누어 구성하였다. 벤치프레스 데이터 세트는 668개의 훈련 데이터와 287개의 테스트 데이터로 나누어 구성하였다. 다음으로 스쿼트 데이터 세트는 279개의 테스트 데이터로 나누어 구성하였다. 마지막으로 데드리프트 데이터 세트는 798개의 훈련 데이터와 342개의 테스트 데이터로 나누어 구성하였다.

4.2 실험 환경

본 실험을 위한 서버는 Windows 11 Pro OS와 프로세서 13th Gen Intel(R) Core(TM) i5-13600KF 3.50GHz, 그래픽 카드 GeForce RTX 4080, 32GB RAM으로 구성된 PC를 사용하고, 클라이언트로 macOS 14와 m1 chip이 탑재된 맥북 m1과 아이폰 12 Pro를 사용하여 실험하였다. 그리고 서버에 사용된 파이썬은 Python 3.10 버전이 사용되었다.

4.3 성능 평가

4.3.1 YOLOv5를 이용한 운동 수행 객체 탐지 모델

본 논문에서 제안한 운동을 수행 중인 사람 객체의 관절을 추정하는 알고리즘을 구현하기 위해 구현한 운동 수행 객체 탐지 모델은 2가지 모델로 구현하였다. 첫 번째로 구현한 모델은 사람 객체의 머리부터 발끝까지 최대한 여백 없이 라벨링한 데이터로 훈련한 모델이며, 두 번째로 구현한 모델은 사람 객체의 머리부터 발끝까지 라벨링하는 것은 동일하지만 상하좌우로 충분한 여백을 포함하여 라벨링한 데이터로 훈련한 모델이다.

본 실험에서 구현한 모델의 성능을 정량적으로 파악하기 위해 사용한 성능 지표로는 컴퓨터 비전 분야에서 자주 사용되는 평균 정밀도(mAP)를 사용하였다. 평균 정밀도는 정밀도(precision)와 재현율(recall)로 구해진다. 정밀도는 검출된 객체의 결과가 얼마나 정확한지를 나타내고 재현율은 대상 객체를 얼마나 잘 감지했는지를 나타내는 지표이다. 평균 정밀도는 모든 클래스에 대한 정밀도와 재현율을 평균내어 계산한다. 따라서 평균 정밀도는 다중 클래스 분류 문제에서 모델의 성능을 종합적으로 측정할 때 흔히 사용되며, 클래스별로 성능이 다를 때 이를 반영하는 중요한 지표이다.

표 10의 그림 21은 첫 번째 모델을 학습시킨 데이터의 라벨링 화면이며 이러한 방식으로 얻은 데이터 세트를 통해 학습시킨 모델을 YOLOv5s를 이용하여 훈련한 운동 수행 객체 탐지 모델의 검증 세트에 대한 객체 탐지 성능을 표 10과 같이 확인하였다.

실험 결과, 임계값(IoU) 0.5 이상일 때 평균 정밀도(mAP) 값은 0.994로 매우 높은 성능을 보였으며 임계값 0.5~0.95 기준으로는 평균 정밀도 0.843의 성능을 보였다. 그러나 이 모델을 통해 감지된 객체의 바운딩 박스 내부의 객체에 대해서만 관절점 추정을 진행한 결과, 오히려 감지의 어려움이 생겨 성능이 저하되는 어려움이 있었다. 이를 해결하고자 표 7의 그림 21과 같이 대상 객체로부터 충분한 여백을 주면 성능이 개선될 것이라 가설을 세우고 다시 라벨링하고 학습을 진행하였다.

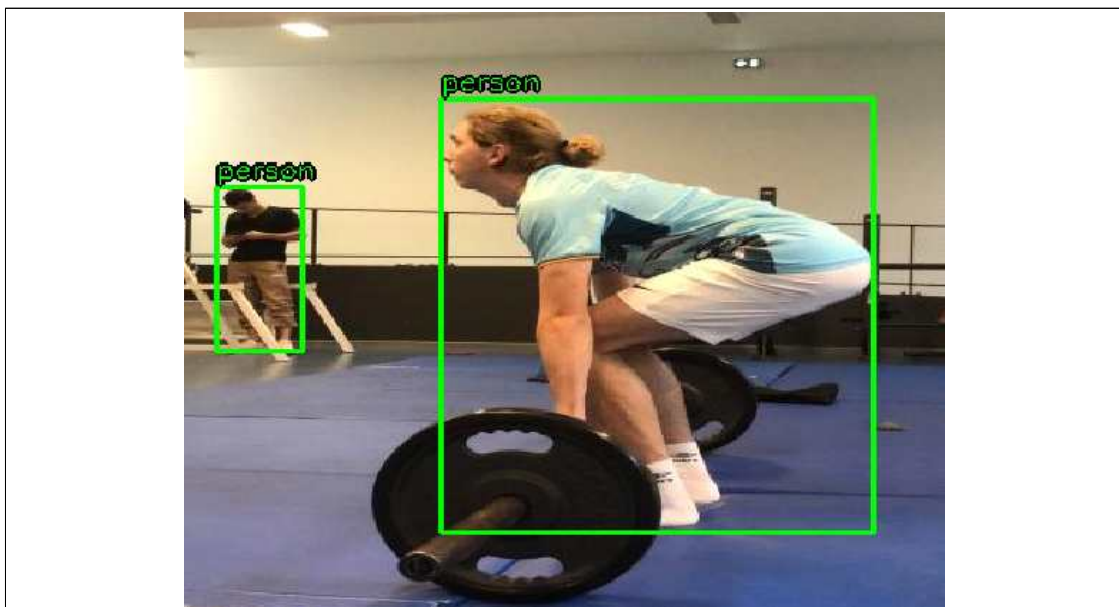


그림 21. 좁은 여백의 라벨링 모습

Precision	Recall	mAP_0.5	mAP_0.5:0.95
0.995	0.984	0.994	0.843

표 10. 여백이 거의 없도록 훈련 데이터를 라벨링한 모델

표 11의 그림 22는 두 번째 모델의 검증 세트에 대한 추론 화면이며 이와 같은 방식으로 충분히 여백을 지정하여 얻은 데이터 세트를 통해 YOLOv5s를 이용하여 훈련한 운동 수행 객체 탐지 모델의 검증 세트에 대한 객체 탐지 성능을 표 11과 같이 확인하였다. 실험 결과, 임계값(IoU) 0.5 이상일 때 평균 정밀도(mAP) 값은 0.994로 첫 번째 모델과 동일하였으며 임계값 0.5~0.95 기준으로는 평균 정밀도 0.705의 성능으로 기존의 모델보다 0.138만큼 감소하였

다. 그러나 이전 모델과는 다르게 그림 21과 같이 바운딩 박스 내부 객체의 관절점을 높은 정확도로 추론하는 결과를 얻을 수 있었다.

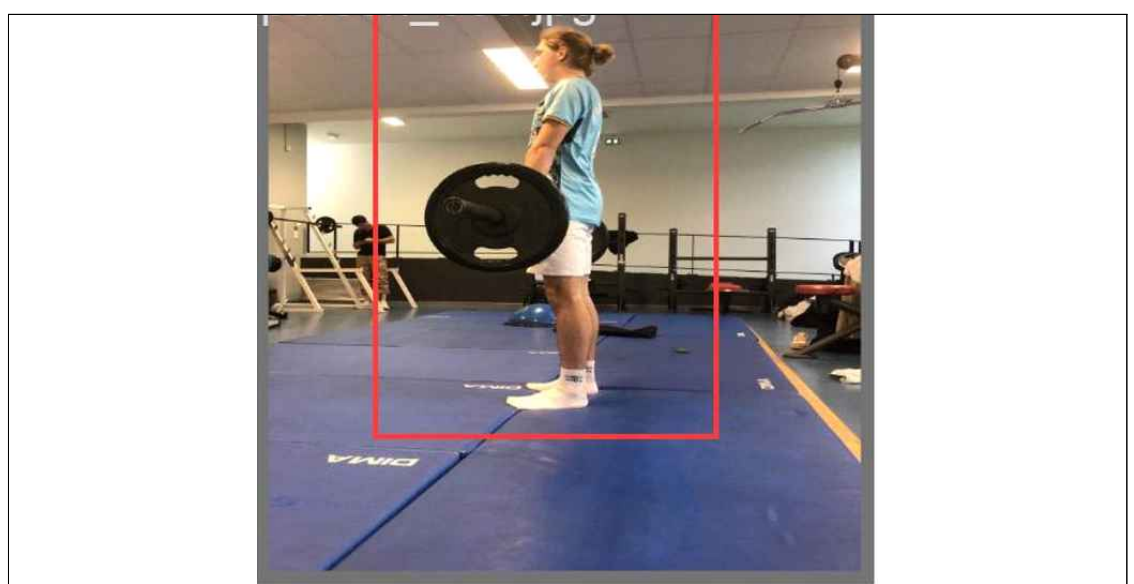


그림 22. 충분한 여백의 라벨링 모습

Precision	Recall	mAP_0.5	mAP_0.5:0.95
0.989	1	0.994	0.705

표 11. 충분한 여백을 포함하여 훈련 데이터를 라벨링한 모델

관절점 추적 신뢰도 임계값

0.70

0.001.00

현재 카운터: 11회

목 각도: 44.66°

왼쪽 어깨 각도: 53.08°

오른쪽 어깨 각도: 60.30°

왼쪽 팔꿈치 각도: 140.92°

오른쪽 팔꿈치 각도: 17.80°

왼쪽 엉덩이 각도: 154.64°

오른쪽 엉덩이 각도: 105.79°

왼쪽 무릎 각도: 96.09°

오른쪽 무릎 각도: 166.41°

왼쪽 발목 각도: 136.74°

오른쪽 발목 각도: 174.35°

실행 중

RUNNING... Stop Deploy

실시간 3대 운동 AI 자세 교정 서비스

운동 선택

벤치프레스



올바른 자세로 운동을 수행하고 있습니다.

올바른 자세로 운동을 수행하고 있습니다.

그림 23. 운동을 수행 중인 사람 객체 탐지 후 관절 좌표 추정(벤치프레스)



그림 24. 운동을 수행 중인 사람 객체 탐지 후 관절 좌표 추정(스쿼트)



그림 25. 운동을 수행 중인 사람 객체 탐지 후 관절 좌표 추정(데드리프트)

4.3.2 운동 자세 분류 모델

본 논문에서 운동 자세 분류 모델의 성능에 대한 정확한 검증을 위해 분류 모델의 성능 평가 지표인 Accuracy, Precision, Recall, F1-Score를 사용하여 각 운동별 자세 분류 모델의 성능 평가를 진행하였다.

표 13은 벤치프레스 수행 시 up과 down 자세에 대해 옳은 자세와 부적절한 자세를 분류하는 모델의 성능 평가 지표 결과 중 F1-Score가 가장 높은 랜덤 포레스트 알고리즘으로 훈련

한 모델의 결과이다. 287개의 테스트 세트를 이용하여 데이터 분류를 진행한 결과, Accuracy는 0.96이 나왔고 본 논문의 목적과 맞는 Weighted avg의 결과로 분류 지표를 해석하였을 때 Precision, Recall, F1-Score 모두 0.96이 나왔다.

벤치프레스	Accuracy	Precision	Recall	F1-Score
Logistic Regression	0.944	0.945	0.944	0.944
Ridge Classifier	0.941	0.942	0.941	0.941
Random Forest	0.962	0.963	0.962	0.962
Gradient Boosting	0.944	0.946	0.944	0.944

표 12. 머신러닝 알고리즘별 벤치프레스 자세 분류 모델 성능 비교

벤치프레스	Precision	Recall	F1-Score	Support
b_correct_up	0.95	1.00	0.97	37
b_correct_down	1.00	0.94	0.97	48
b_excessive_arch_up	0.95	1.00	0.98	40
b_excessive_arch_down	0.98	0.96	0.97	51
b_arms_spread_up	0.91	0.98	0.94	51
b_arms_spread_down	0.98	0.92	0.95	60
Accuracy			0.96	287
Macro avg	0.96	0.97	0.96	287
Weighted avg	0.96	0.96	0.96	287

표 13. 랜덤 포레스트 알고리즘으로 학습한 벤치프레스 자세 분류 모델의 성능 지표 결과

표 15는 스쿼트를 수행 시 up과 down 자세에 대해 옳은 자세와 부적절한 자세를 분류하는 모델의 성능 평가 지표 결과 중 F1-Score가 가장 높은 랜덤 포레스트 알고리즘으로 훈련한 모델의 결과이다. 279개의 테스트 세트를 이용하여 데이터 분류를 진행한 결과, Accuracy는 0.99가 나왔고 본 논문의 목적과 맞는 Weighted avg의 결과로 분류 지표를 해석하였을 때 Precision, Recall, F1-Score 모두 0.99가 나왔다.

스쿼트	Accuracy	Precision	Recall	F1-Score
Logistic Regression	0.986	0.986	0.989	0.986
Ridge Classifier	0.986	0.986	0.986	0.986
Random Forest	0.989	0.989	0.989	0.989
Gradient Boosting	0.982	0.982	0.982	0.982

표 14. 머신러닝 알고리즘별 스쿼트 자세 분류 모델 성능 비교

스쿼트	Precision	Recall	F1-Score	Support
s_correct_up	1.00	1.00	1.00	42
s_correct_down	1.00	1.00	1.00	30
s_spine_neutral_up	0.97	0.94	0.95	33
s_spine_neutral_down	0.94	0.97	0.95	30
s_caved_in_kness_up	1.00	1.00	1.00	41
s_caved_in_kness_down	1.00	1.00	1.00	38
s_feet_spread_up	1.00	1.00	1.00	36
s_feet_spread_down	1.00	1.00	1.00	29
Accuracy			0.99	279
Macro avg	0.99	0.99	0.99	279
Weighted avg	0.99	0.99	0.99	279

표 15. 랜덤 포레스트 알고리즘으로 학습한 스쿼트 자세 분류 모델의 성능 지표 결과

표 17은 데드리프트를 수행 시 up과 down 자세에 대해 옳은 자세와 부적절한 자세를 분류하는 모델의 성능 평가 지표 결과 중 F1-Score가 가장 높은 랜덤 포레스트 알고리즘으로 훈련한 모델의 결과이다. 342개의 테스트 세트를 이용하여 데이터 분류를 진행한 결과, Accuracy는 0.95가 나왔고 본 논문의 목적과 맞는 Weighted avg의 결과로 분류 지표를 해석하였을 때 Precision, Recall, F1-Score 모두 0.95가 나왔다.

데드리프트	Accuracy	Precision	Recall	F1-Score
Logistic Regression	0.898	0.900	0.898	0.897
Ridge Classifier	0.874	0.882	0.874	0.873
Random Forest	0.947	0.949	0.947	0.942
Gradient Boosting	0.942	0.944	0.942	0.942

표 16. 머신러닝 알고리즘별 데드리프트 자세 분류 모델 성능 비교

데드리프트	Precision	Recall	F1-Score	Support
d_correct_up	0.98	0.93	0.95	57
d_correct_down	1.00	0.97	0.99	35
d_spine_neutral_up	0.92	0.92	0.92	52
d_spine_neutral_down	0.89	0.93	0.91	44
d_arms_spread_up	1.00	0.94	0.97	32
d_arms_spread_down	0.88	1.00	0.94	38
d_arms_narrow_up	0.91	0.95	0.93	44
d_arms_narrow_down	0.95	0.88	0.91	40
Accuracy			0.95	342
Macro avg	0.95	0.95	0.95	342
Weighted avg	0.95	0.95	0.95	342

표 17. 랜덤 포레스트 알고리즘으로 학습한 데드리프트 자세 분류 모델의 성능 지표 결과

5. 결론

본 논문에서는 MediaPipe의 Pose Estimation 기술을 이용하여 관절점(landmarks)을 추출하고 그 결괏값을 사용하여 머신러닝 모델을 구축하였다. 이를 OpenCV와 Streamlit을 활용하여 웹 브라우저 환경에서 사용자의 3대 운동 자세를 실시간으로 추적하고 학습시킨 모델을 기반으로 자세를 분류하며 잘못된 자세에 대해서는 음성으로 피드백을 제공한다.

MediaPipe는 GPU가 존재하지 않거나 사양이 낮은 Mobile Device나 PC의 Web Cam과 같이 CPU와 카메라가 있는 환경이라면 실시간 자세 추정이 가능하다. 하지만 이 프레임워크는 단일 인물에 대해서만 Pose Estimation이 가능하다는 단점이 있는데 본 논문에서는 이를 YOLOv5를 활용하여 운동을 수행 중인 사람 객체 탐지 모델을 구현하여 실제 운동하는 사람 객체만 탐지하여 Pose Estimation의 정확성을 높였다.

본 논문에서 제안하는 YOLOv5 및 MediaPipe를 이용한 실시간 3대 운동 AI 자세 교정 서비스는 단순히 정적인 이미지에 대해서 피드백하는 것이 아니라 운동을 수행하는 동적인 움직임을 매 프레임마다 분석하며 사용자가 정확하게 운동을 진행하는지 확인하고 올바르게 않은 자세에 대해서 음성합성 기술(TTS)을 활용해 적절한 피드백을 제공한다는 점에서 웨이트 트레이닝을 안전하게 수행할 수 있도록 하는 데 기여할 수 있을 것으로 기대된다.

본 논문에서는 학습에 사용한 데이터 수가 적으며 머신러닝 기법인 랜덤 포레스트 알고리즘만을 활용하여 구현하였다는 한계점이 있다. 향후 연구로써, 실험자의 데이터뿐만 아니라 더 많은 데이터를 이용하고 딥러닝 모델을 이용하여 구현할 경우, 더 다양한 환경에서 높은 정확도로 자세 분류를 할 수 있을 것으로 예상된다.

참고문헌

- [1] 문화체육관광부, 2022 국민생활체육조사, 2023. 09. 03 검색,
https://www.mcst.go.kr/kor/s_policy/dept/deptView.jsp?pSeq=1691
- [2] 이용준, 김태영, "딥러닝 기반 포즈 인식 및 교정을 통한 효율적인 홈 트레이닝 시스템 개발", 2021.
- [3] 신윤지, 이현주, 김준희, 권다영, 이선애, 추윤진, 박지혜, 정자현, 이형석, 김준호, "딥러닝 기반 영상처리 기법 및 표준 운동 프로그램을 활용한 비대면 온라인 홈트레이닝 어플리케이션 연구", 2021.
- [4] S. Shirsath, P. Thakkar, B. Wolfson, "Using Human Pose Detection to Identify and Give Feedback on Workout Form"
- [5] R. R. Kanase, A. N. Kumavat, R. D. Sinalkar, S. Somani, "Pose Estimation and Correcting Exercise Posture", 2021.
- [6] Swain, D.; Satapathy, S.; Acharya, B.; Shukla, M.; Gerogiannis, V.C.; Kanavos, A.; Giakovis, D. Deep Learning Models for Yoga Pose Monitoring. Algorithms 2022
- [7] 김규원, 이주환, 최보훈, 홍성민, 유철중, 김순태. "피트니스 자세 교정을 위한 MediaPipe 기반 운동 분석 서비스 아키텍처". 한국정보기술학회 종합학술발표논문집, 664-669. 2023.
- [8] Ultralytics, YOLOv5, [Internet], <https://github.com/ultralytics/yolov5>
- [9] C. Lugaresi, et al. "Mediapipe: A framework for building perception pipelines," in Proceedings of Third Workshop on Computer Vision for AR/VR at IEEE Computer Vision and Pattern Recognition, Long Beach, June 2019.
- [10] 김윤상, 김고은, 문채은, 이나경, 김웅식. "뇌파 데이터를 활용한 GRU 모델의 인간 감정 분류 성능 평가에 관한 연구". 한국통신학회 학술대회논문집, 356-357. 2023
- [11] "Squat-Depth Image Dataset", roboflow. last modified Aug, 2023, accessed Sep 10, 2023
- [12] "SDT Image Dataset", roboflow. last modified May, 2023, accessed Sep 10, 2023
- [13] "HumonBody1 Computer Vision Project", roboflow. last modified Nov, 2022, accessed Sep 10, 2023
- [14] "Faller Image Dataset", roboflow. last modified Jul, 2023, accessed Sep 10, 2023
- [15] "Silhouettes of human posture", kaggle. last modified Jul, 2020, accessed Sep 14, 2023

요약서

최근 들어 웨이트 트레이닝의 관심도가 증가하고 있으며 이에 따라 헬스장에서 웨이트 트레이닝을 하는 사람들이 늘어나고 있다. 웨이트 트레이닝의 핵심에는 벤치프레스, 스쿼트, 데드리프트를 합쳐 부르는 3대 운동이 있다. 3대 운동의 경우 잘못된 자세로 운동을 수행하게 되면 큰 부상을 야기할 수 있다. 본 논문은 YOLOv5 및 MediaPipe를 이용한 실시간 3대 운동 AI 자세 교정 서비스를 제안한다. 본 논문은 YOLOv5로 구현한 운동을 수행하는 사람 객체를 탐지하는 모델을 이용하여 운동을 수행 중인 사람 객체를 탐지하고 해당 객체에 대해서 MediaPipe의 Pose Estimation 기술을 활용하여 관절점을 추출한다. 추출된 관절점 데이터를 세트로 학습하여 운동 수행 자세 분류 모델을 구성한다. 이를 OpenCV와 Streamlit을 활용하여 웹 브라우저에서 실행할 수 있으며 카메라로부터 입력받은 비디오 데이터를 매 프레임마다 분석하여 잘못된 자세에 대해서는 음성합성(TTS) 기술을 활용하여 피드백을 제공한다. 또한 운동을 수행할 때마다 현재 자세를 파악하여 'up'과 'down'으로 상태를 파악하여 운동 횟수를 세도록 하여 보다 효율적인 운동이 가능하게 한다. 랜덤 포레스트 알고리즘으로 구현한 벤치프레스, 스쿼트, 데드리프트 자세 분류 모델의 F1-Score는 각각 0.96, 0.99, 0.95로 높은 성능을 보였다는 것을 확인하였다.