

YOLOv7와 ML Kit 기반 안드로이드용 운동 자세 분석 어플리케이션

박희완*, 송현교^o

*한라대학교 IT소프트웨어학과 부교수,

^o한라대학교 IT소프트웨어학과 학부생

e-mail: heewanpark@halla.ac.kr*, songhg0321@naver.com^o

YOLOv7 and ML Kit based Weight Training Posture Analyse Application for Android

Heewan Park*, Hyeongyo Song^o

*Associate Professor of Dept. of IT Software, Halla University,

^oUndergraduate Student of Dept. of IT Software, Halla University

● 요약 ●

본 논문에서는 YOLOv7-tiny와 ML Kit를 안드로이드 어플리케이션에 통합함으로써 사용자의 웨이트 트레이닝 자세 개선에 도움을 주는 안드로이드용 Bar Path Tracking 및 포즈 추정 기술의 구현 과정에 관하여 제안한다. 사용자가 영상을 선택하면 영상의 프레임 추출 작업 및 추출 프레임에 대한 객체 탐지와 포즈 추정 작업이 수행되며, 자세 및 시간의 흐름에 따른 바벨의 이동 경로가 매우 직관적으로 표시된 결과 영상이 사용자에게 반환된다. 이 기술을 통해 사용자는 휴대가 용이한 자신의 스마트폰으로 언제 어디서나 숙련자들의 트레이닝 영상을 분석할 수 있으며, 이를 자신의 트레이닝에 적용함으로써 전문가의 지도를 받을 수 없는 상황에서도 쉽고 빠르게 자세의 개선을 이룰 수 있을 것이라 기대한다.

키워드: 안드로이드(Android), 객체탐지(Object Detection), 포즈추정(Pose Estimate)

I. Introduction

웨이트 트레이닝에서 정확한 자세를 유지하는 것은 부상 방지와 근육 성장에 직결된다. 본 연구에서는 트레이닝 자세의 개선을 돕기 위해, 전문가의 지도 없이도 자세를 직관적으로 확인할 수 있게 해주어 자가 트레이닝의 효과를 극대화시키는 어플리케이션을 개발하였다. 트레이닝 종목의 대부분은 정형화된 자세가 존재하므로, 바벨 궤적(Bar Path)의 정석[1]을 숙지하여 이를 흉내 내는 것만으로 쉽고 빠르게 자세의 개선을 이룰 수 있다.

세계 최초의 스마트폰 'IBM 사이먼'[2]이 등장한 지 약 30년가량밖에 지나지 않았음에도 오늘날의 스마트폰은 작은 컴퓨터라 불릴 정도의 엄청난 휴대성과 성능을 자랑한다. 이에 인구 대부분이 스마트폰을 소지한 상태이며 사용자의 편의성을 위한 모바일 어플리케이션들은 끊임없이 제안되고 있다.

그 중 피트니스 시장에는 운동 플래너, 운동 타이머, 3D 모델링을 활용한 트레이닝 자세 교본 등 다양한 트레이닝 보조 어플리케이션들이 존재하지만, 객체 탐지와 포즈 추정을 통합하여 바벨의 움직임 및 포즈를 직관적으로 볼 수 있게 해주는 영상 분석 어플리케이션이나 관련 연구가 매우 드문 상태이다.

본 어플리케이션은 약간의 자세 개선만으로도 중량 상승을 빠르게 이룰 수 있는 초보자부터, 자세의 개선이 있어야만 중량 상승을 이룰 수 있는 웨이트 트레이닝 숙련자까지 폭넓은 수준에서 활용 가능하다. 특히 무거운 무게를 드는 것을 목적으로 하는 역도 선수나 파워리프터, 올바르지 않은 자세 탓에 부상을 경험한 적 있어 자세 교정이 필요한 자들에게 더욱 적합할 것으로 기대한다.

II. Preliminaries

1. Related works

1.1 국내 동향

‘바벨 궤적 추적 및 운동 자세 분석을 통한 웨이트 트레이닝 보조 시스템’[3] 논문이 유사 연구가 서술된 바 있다. 해당 논문에서는 YOLOv3와 OpenCV 인공지능 딥러닝을 활용하여 사용자가 촬영한 바벨 웨이트 트레이닝 영상을 분석하고, 분석의 결과로 바벨의 궤적뿐만 아니라 포즈 추정을 통해 관절의 이격까지 시각화함으로써 사용자

에게 직관적인 자세 피드백을 제공하는 시스템에 관해 서술하였다. 즉 객체 탐지와 포즈 추정이 모두 통합된 시스템이며 본 연구에서 제시하는 기술보다 향상된 트레이닝 보조 기술을 보이지만 모바일이 아닌 데스크탑 환경에서만 작동하는 시스템이라는 점에서 본 연구와 차이가 있다.

1.2 유사 어플리케이션

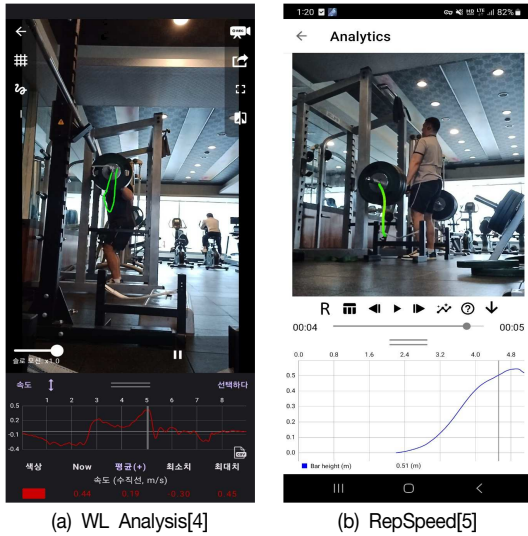


Fig. 1. Related Applications

Fig. 1은 현재 Google Play Store에 존재하는 두 개의 Bar Path Tracking 어플리케이션 동작 모습을 캡처한 것이다. 해당 어플리케이션들은 객체 탐지 AI를 활용하여 바벨의 경로를 정확하게 시각화하고 바벨의 이동 속도를 그래프로 표현해 준다는 특징을 가졌다. 그러나 포즈 추정을 통한 관절의 움직임을 시각화하는 기능은 접목되지 않았다는 점에서 본 연구와 차이가 존재한다.

III. The Proposed Scheme

본 연구의 기본적인 아이디어는 YOLO[6] 모델이 영상을 인자로 받을 경우 해당 영상의 모든 프레임을 순회하며 객체 탐지 정보를 출력하는 처리 과정을 반복한다는 점에서 착안하였다.

YOLO는 AI의 학습과 사용에 최적화된 PyTorch[7]를 기반으로 구현되었기 때문에 코드를 직접 이식하는 것은 불가능할 것으로 판단하였고, 이를 안드로이드 환경에서 최대한 유사하게 구현해 내기 위한 대안으로 영상을 프레임 단위로 분해하는 기능과 여러 장의 프레임들을 이어 붙이는 기능을 가진 도구 중 Apache License 2.0으로 라이선스가 유연한 MediaExtractor API[8]와 MediaMuxer API[9]를 채택하게 되었다.

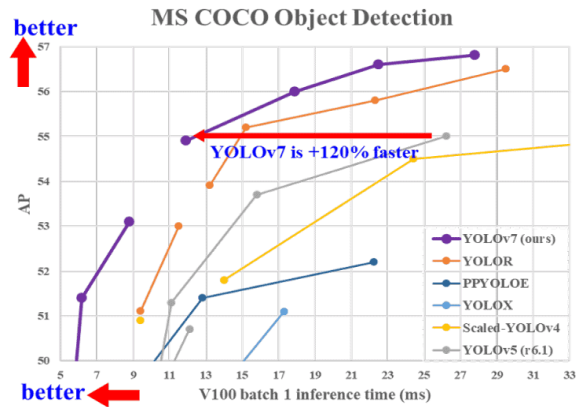


Fig. 2. Different results of MS COCO Object Detection[10]

Fig. 2는 MS COCO Dataset[10]에 대하여 여러 버전의 YOLO 모델 중 YOLOv7이 가장 빠른 추론 속도를 보였다는 연구 그래프를 발췌한 것이다[11]. 연구 당시 객체 탐지 모델의 선정에 위해 Fig. 2를 적극 참고하였으며, 오랜 시간 다양한 분야에서 사용되며 많은 검증 단계를 거쳐 온 YOLOv7이 최신 YOLO 모델들에 비해 높은 안정성을 보일 것이라 판단하였다. 작동 환경이 모바일 환경이라는 점을 고려해 YOLOv7의 경량화 모델인 YOLOv7-tiny를 선택하여 추론 속도를 높였다. 또한 포즈 추정을 위해, On-Device 모드를 지원하며 선행 학습된 모델을 제공하는 Google의 ML Kit[12]를 이용하여 구현에 필요한 자원을 절약하였다.

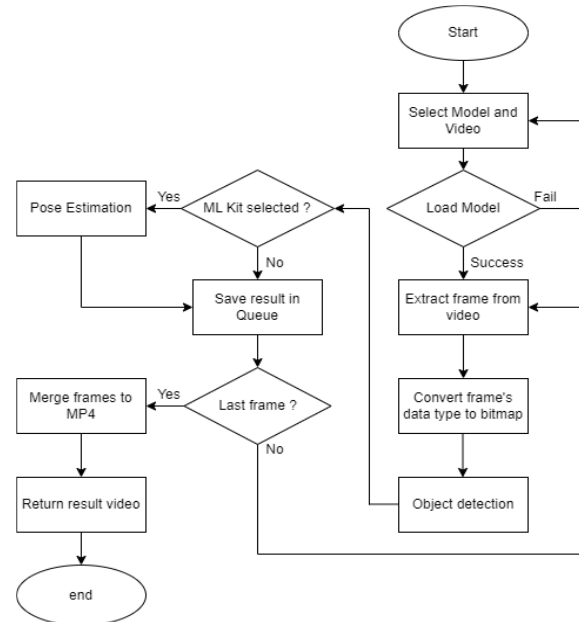


Fig. 3. System Flow Chart

본 연구는 Fig. 3와 같이 구현되었다. 특정 물체에 의해 신체가 가려질 시 포즈 추정이 잘못된 결과를 출력하는 경우가 존재하며 이러한 현상은 주로 신체가 가려지기 쉬운 완전한 측면 시점에서 두드러지기 때문에, 촬영된 영상의 구도에 맞게 포즈 추정 수행 여부를 사용자가 직접 결정할 수 있는 ToggleButton을 두어 Toggle 하지

않을 시 객체 탐지만 수행되도록, Toggle 시 객체 탐지와 포즈 추정이 함께 수행되도록 하였다. 그 아래에는 영상 분석 시작의 트리거로, 안드로이드 미디어 저장소에 접근하여 영상을 선택하기 위한 버튼을 두었다.

사용자의 영상 선택이 완료되면 YOLOv7-tiny와 ML Kit Pose Estimate 모델이 기기에 Load되며, 두 모델이 성공적으로 Load 될 시 MediaExtractor API를 통한 영상 프레임 추출이 시작되도록 하였다.

추출된 프레임은 Image 타입으로, 직접적인 픽셀 데이터 수정이 불가능해 객체 탐지 결과를 프레임 상에 그려낼 수 없다는 문제가 존재한다. 따라서 이를 수정이 용이한 Bitmap 타입으로 변환하는 과정을 거칠 필요가 있었으며 적절한 API나 메서드가 존재하지 않는 관계로 Image의 ByteBuffer값을 추출하여 Bitmap 타입으로 변환하는 ImageToBitmap 메서드를 직접 구현함으로써 문제를 해결하였다.

Bitmap 타입으로 변환된 프레임은 Java Native Interface[13]를 통해 YOLOv7-tiny 관련 메서드들을 정의해둔 C++ 파일로 전달되며, 파일 내 객체 탐지 메서드가 이를 인자로 받아 탐지를 수행하고 그 결과를 다시 Bitmap 타입으로 반환하게 하였다.

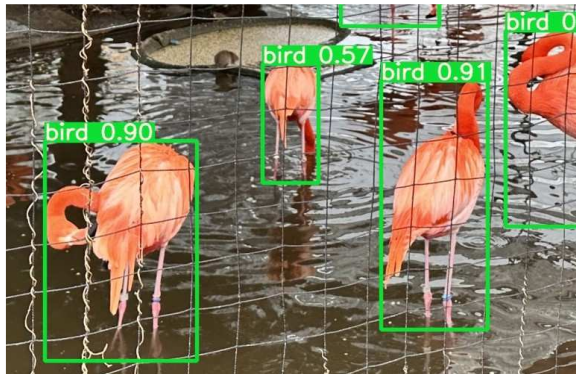


Fig. 4. YOLOv7's Bounding Box

이때 Fig. 4처럼 바운딩 박스를 표시하는 일반적인 결과 출력 방식과 달리, 감지된 객체의 중심 좌표에 빨간색의 점을 찍도록 설정하였다. 또한 프레임마다 객체의 중심 좌표를 배열에 저장하고, 다음 프레임을 처리할 때 배열 내에 존재하는 모든 이전 좌표에 한 번 더 점을 찍도록 하여 시간의 흐름에 따른 바벨의 위치 변화와 속도를 직관적으로 볼 수 있도록 하였다. 또한 점이 찍힌 후 일정 시간이 지나면 색상이 회색으로 변경되도록 설정하여 가시성을 더욱 향상시켰다.

객체 탐지 이후, 최초로 사용자가 ToggleButton을 활성화 하였는지 여부를 검사한다. 활성화되었다면 객체 탐지 결과로 반환된 Bitmap을 ML Kit 관련 메서드가 인자로 받아 포즈 추정까지 수행한 뒤 결과 Bitmap을 반환하게 하였으며, 활성화되지 않았다면 포즈 추정을 수행하지 않고 넘어가도록 하였다.

분석이 완료된 Bitmap은 Queue에 삽입되며, 영상의 프레임 추출부터 분석이 완료된 프레임을 Queue에 삽입하는 일련의 과정을 루프의 한 사이클로 규정하여 사이클 완료 시 루프의 시작 부분으로 다시 돌아가 동일한 과정을 모든 프레임에 대한 처리가 완료될 때까지

반복하게 하였다. 루프 종료 후 최종 단계에서 MediaMuxer API를 통해 Queue의 프레임들을 순차적으로 꺼내며 합쳐나가 최종적으로는 하나의 MP4 영상으로 저장하여 사용자에게 반환하는 기술의 구현에 성공하였다.

Windows 10, 16GB RAM, NVIDIA GeForce GTX 1060 3GB, Android Studio 환경에서 JAVA와 C++ 언어를 사용하여 어플리케이션을 개발하였으며, 동일 환경에서 바벨의 끝점만 집중적으로 Labelling한 3136장의 Training 이미지, 395장의 Validation 이미지, 143장의 Test 이미지를 1500 Epochs로 설정하여 YOLOv7-tiny 모델의 학습을 진행하였다.

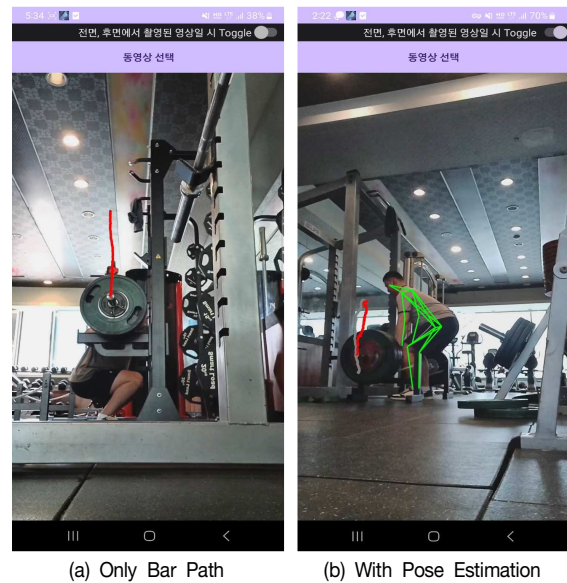


Fig. 5. Result of Implementation

Fig. 5는 연구 결과를 기반으로 구현한 실제 어플리케이션의 동작 모습 일부를 캡처한 것이다. 액티비티 상단에 포즈 추정 활성화를 위한 ToggleButton이 존재함을 확인할 수 있다. ToggleButton이 활성화되지 않은 경우 (a)와 같이 객체 탐지만이 수행되고, 활성화된 경우 (b)와 같이 객체 탐지와 포즈 추정이 함께 수행된다. Fig. 5를 통해 결과가 가시성 높은 색상으로 잘 표현되고 있음을 확인할 수 있다.

IV. Conclusions

현대 기술의 집합체인 스마트폰은 사회에 깊은 뿌리를 내려 일상생활에 없어서는 안 될 존재로 자리매김하였다. 이에 다양한 분야에서 사용자를 위한 모바일 어플리케이션을 개발하려는 노력이 지속되고 있으나 동작 환경의 성능에 크게 의존하는 AI의 특성 상 모바일 환경에 AI를 접목한 사례가 데스크탑에 비해 상대적으로 적다고 판단하였다. 이러한 이유로 본 연구에서는 모바일과 AI의 결합에 집중하였으며, 그 시작으로 이번 트레이닝 자세 교정 어플리케이션 관련 연구를 계획하였다.

YOLO 모델의 영상 처리 과정을 모티브로 하였으며, 이를 모바일에

서 유사하게 구현하기 위해 적절한 API 선정과 메서드 구현 과정을 거쳐 AI가 바벨의 끝부분과 인간의 관절을 인식하고 가시성 높은 형태로 시각화하여 사용자에게 제공하는 안드로이드 어플리케이션의 구현에 성공하였다.

유사 시스템의 대부분이 데스크탑용으로 구현되었거나, 모바일 환경에서 동작하더라도 실시간 감지만을 지원하거나, 객체 탐지에만 초점을 두었다. 그러나 본 연구는 모바일 환경에 객체 탐지와 포즈 추정을 모두 통합함으로써, 다소 이해하기 어려울 수 있는 그래프 대신 웨이트 트레이닝 초보자도 한눈에 알아볼 수 있는 디자인을 사용해 휴대성과 직관성을 최대화하였다는 점에서 의의가 있다.

이를 통해 사용자는 휴대가 용이한 자신의 스마트폰으로 언제 어디서나 숙련자들의 트레이닝 영상을 분석하고 그것을 자신의 트레이닝에 적용함으로써 전문가의 지도를 받을 수 없는 상황에서도 스스로 트레이닝 자세를 개선할 수 있을 것이라 기대할 수 있다.

[13] Java Native Interface, <https://developer.android.com/training/articles/perf-jni?hl=ko>

REFERENCES

- [1] H. Nagao, Y. Kubo, T. Tsuno, S. Kurosaka and M. Muto, "A Biomechanical Comparison of Successful and Unsuccessful Snatch Attempts among Elite Male Weightlifters," Sports, Vol. 7, No. 6, pp. 151, 2019.
- [2] IBM Simon, https://en.wikipedia.org/wiki/IBM_Simon
- [3] Seongjae Bae, Joonsuk Oh, and Rhan Ha, "A Weight-training Assistance System with Bar Path Tracking and Posture Analysis," Proceedings of KIIT Conference, pp. 711-715, Jeju, 2022.
- [4] WL Analysis, <https://play.google.com/store/apps/details?id=com.karolsmolak.wlanalysis>
- [5] RepSpeed, <https://play.google.com/store/apps/details?id=com.sire.repspeed>
- [6] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You only look once: unified, real-time object detection," Proceedings of the IEEE conference on computer vision and pattern recognition, pp 779-788, 2016.
- [7] PyTorch, <https://pytorch.org/>
- [8] MediaExtractor, <https://developer.android.com/reference/android/media/MediaExtractor>
- [9] MediaMuxer, <https://developer.android.com/reference/android/media/MediaMuxer>
- [10] MS COCO Datasets, <https://cocodataset.org/#home>
- [11] CY. Wang, A. Bochkovskiy and HYM. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 7464-7475, 2022.
- [12] ML Kit, <https://developers.google.com/ml-kit?hl=ko>