

실시간 Key Point 추정을 이용한 학습 도움 시스템에 관한 연구

이혁재, 황태운, 박구만

서울과학기술대학교 전자 IT 미디어공학과

lhj101296@gmail.com, htw0826@naver.com, gmpark@seoultech.ac.kr

Study Helper System using Key Point Estimation

HyukJae Lee, Taeun Hwang, Gooman Park

Seoul National Univ of Science & Technology

Department of Electronic & IT Media Engineering

요 약

본 논문에서는 웹을 이용하여 학습진단을 제공하는 시스템을 제안하였다. 이를 위하여 Pose estimation 과 시선 추적 기법을 적용하여 데이터를 얻고, 자신의 학습에 대한 통계를 학습 시행별로 시각화 하여 보여줌으로써 스스로 진단할 수 있게 하였다. 실험을 통하여 개인이 접근하고 활용하기 어려웠던 자세 추정을 실시간으로, 적은 자원으로 원활히 작동하는 결과를 얻었다.

I. 서 론

최근 들어 비대면 강의와 인터넷 강의가 증가함에 따라 수업을 들으며 자세가 나빠지거나 집중력이 흐트러지는 현상이 증가하고 있다. 부적절한 자세를 취하는 것은 스스로 인지하기 어렵기 때문에 외부의 도움을 받는 방법이 필요하다.

Pose estimation 과 Eye tracking 은 최근 많은 발전을 이루어 Multi-frame, 즉 비디오에서도 양질의 퍼포먼스를 보이기 때문에 활용할 수 있는 범위가 매우 넓어졌다.

위 기술들은 효용성이 매우 높지만 사용하는 데에는 많은 컴퓨팅 자원을 필요로 한다. 우선 본 프로그램의 목적은 일반 사무용 노트북과 같은 환경에서도 알고리즘이 원활하게 동작하도록 하는 것이다. 본 논문에서는 이상의 목적을 달성하여 Pose estimation 과 Eye tracking 을 실시간으로, 적은 컴퓨팅 자원을 이용하여 접근성이 용이한 웹 애플리케이션으로 사용할 수 있는 알고리즘을 제안한다.

II. 키포인트 측정 및 계산, 시스템 구성

1. 거북목 탐지

적은 컴퓨팅 자원으로 Pose estimation 을 사용하기 위한 조사 결과, Movenet 이라는 모델이 적합했다. Movenet 은 lighting 버전과 thunder 버전으로 나뉜다. lighting 버전은 일반 노트북에서 50fps 정도의 퍼포먼스를 보여준다.^[1]

머리자세 판단을 위하여 전방머리자세각도(Craniovertebral Angle, CVA)를 사용한다. CVA 는 경추 7 번의 극돌기 끝부분과 이주(Tragus)를 이은 선이 이루는 각을 의미한다.^[2] 그림 1 에 CVA 를 나타내었다.

CVA 가 50° 미만일 경우에 의학적으로 거북목 증후군으로 판단한다.^[3]

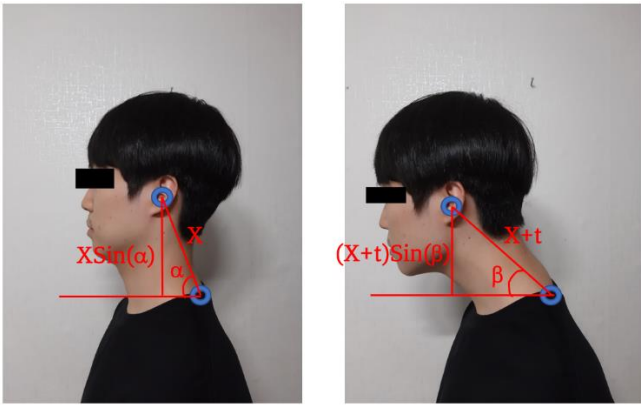


그림 1. 전방머리자세각도 판단 방법

정상 자세일 때 CVA(α)는 68° 내외이다.^[4] 정면자세에서 거북목을 판단하기 위해 수식 1에 Sin을 이용해 길이비를 구하는 식을 표현하였다.

수식 1

$$\frac{(X+t)\sin(50^\circ)}{X\sin(68^\circ)} \approx th$$

본 논문에서는 목 길이 변화는 없다고 가정하여 Threshold = 0.826으로 설정하고, 이 값 미만일 경우 거북목으로 판단한다. 또한 어깨 길이 대비 비율을 사용하여 사용자와 캠 사이의 거리의 변화에 따른 길이 측정의 오류를 해결하였다.

2. 집중도 계산

1) 고개의 방향 파악

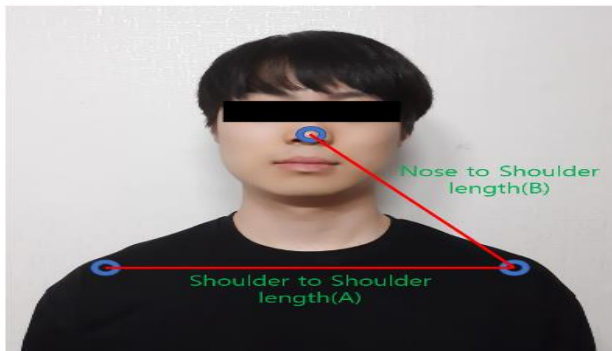


그림 2. 고개의 방향 파악 방법

어깨와 귀의 좌표를 구해 어깨 사이의 거리(A)와 코와 어깨 사이의 거리(B)를 측정하고 이들의 비율을 이용하여 고개의 방향을 알 수 있는 수식을 수식 2로 표현하였다.

수식 2

$$\frac{B}{A} < \rho \rightarrow \text{Left}$$

$$\rho \leq \frac{B}{A} < \sigma \rightarrow \text{Center}$$

$$\sigma \leq \frac{B}{A} \rightarrow \text{Right}$$

본 논문에서는 여러 번의 시뮬레이션으로 $\rho = 0.45$, $\sigma = 0.7$ 을 사용하여 고개 방향 탐지를 통한 집중시간을 측정하였다.

2) 눈동자 방향 파악

Movenet 으로는 눈의 세분화된 Landmark를 얻을 수 없어 이를 제공하는 Mediapipe Iris를 사용하였다.

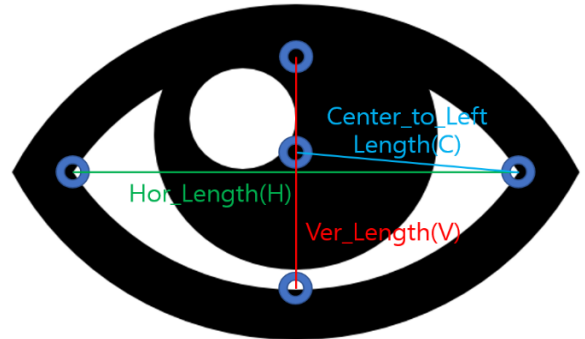


그림 3. 눈동자 방향 파악 방법

그림 3에 표시된 위치의 좌표를 측정하여 눈의 수평길이(H), 눈의 수직길이(V), 왼쪽 끝에서 눈동자 중심까지의 길이(C)를 계산한다. Equation 2에서 B/A 를 C/H 로 바꿔 눈동자의 방향을 탐지하고, 정면을 보지 않는 시간을 측정한다. 이 때 $\alpha = 0.45$, $\beta = 0.54$ 를 사용하였다.

3) 졸음시간, 멍한 시간 측정

그림 3의 H와 V의 비율을 사용해 눈 깜빡임을 측정하고 이 시간을 기록한다. 눈을 깜빡이기 시작한 순간의 기록(First Time Record, FTR), 눈을 깜빡인 마지막 시간기록(Last Time Record, LTR), 현재 시간(Current Time, CT)을 기준시간(standard Time, ST)과 비교하여 졸음 상태를 판단한다. 또한 기준시간동안 눈동자 움직임(M) 변화율을 계산하여 멍한 상태를 판단한다. 이를 수식 3으로 표현하였다.

수식 3

$$\frac{H}{V} \geq \gamma \rightarrow \text{Blinking Time Record}$$

$$CT - FTR > ST \rightarrow \text{Dozing}$$

$$CT - LTR > ST \text{ and } \frac{\Delta M}{\Delta ST} < \varepsilon \rightarrow \text{Spaced Out}$$

거북목, 집중도 탐지 알고리즘을 Flow Chart로 그림 4에 표현하였다.

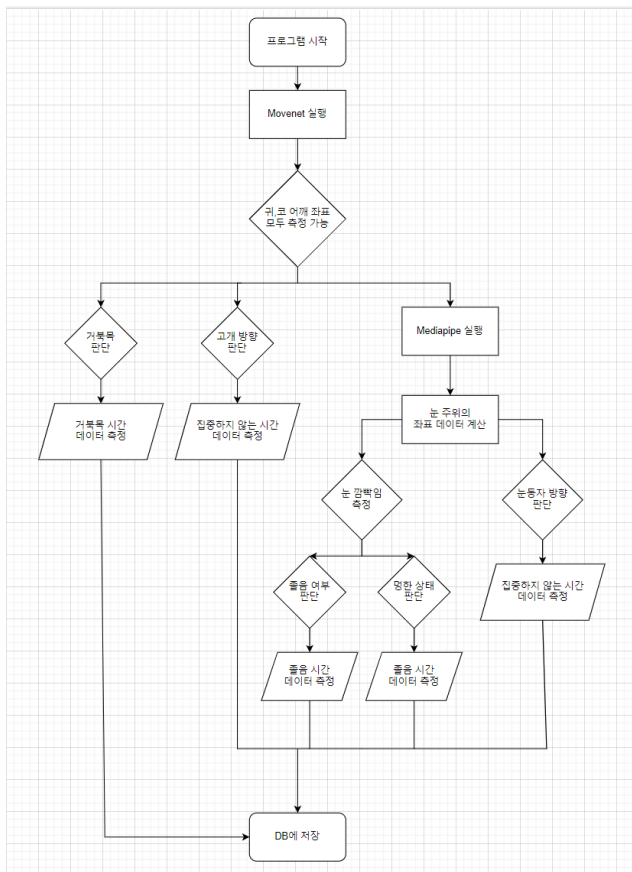


그림 4. Flow Chart

3. 웹 페이지 구성

사용자에 따른 피드백 페이지를 구성함에 있어 Django가 기능면에서 유용하기에 Django로 전체 flow를 구상하였다. URL Mapping을 사용하면 하나의 html 파일을 재사용하여 전혀 다른 데이터가 담긴 페이지를 제공할 수 있기 때문이다

Django는 웹 프레임워크이기 때문에 서버 컴퓨터에서만 작동된다. 사용자에게 Python 파일을 제공하는 것은 불가능하기 때문에 PyScript를 사용한다. Anaconda가 PyCon US 2022에서 PyScript를 발표하였는데, 이는 Python과 표준 HTML을 결합하여 브라우저에서 Python 애플리케이션을 만들 수 있도록 지원하는 프레임워크이다. 이를 이용해 사용자에게 Python 동작을 지원한다. (그림 5)

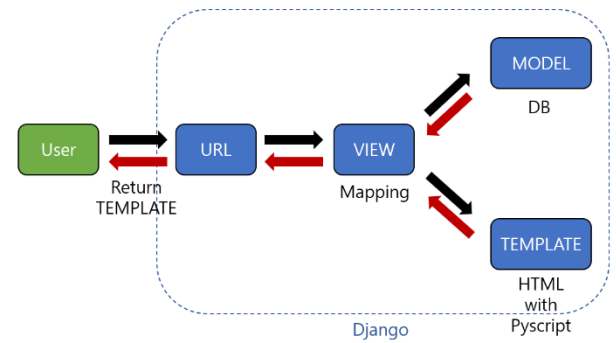


그림 5. Django 시스템 구성도

III. 실험 및 결과고찰

1. 실험 환경

본 논문의 실험 환경은 intel i5-1135G7와 intel Iris Xe Graphics를 가지는 LG 그램 21을 사용하였다.

2. 실험 결과

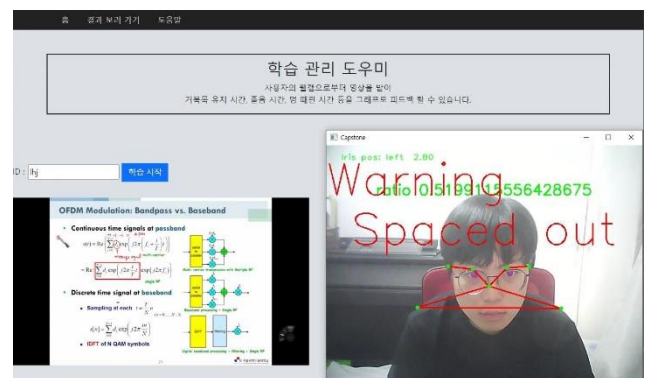


그림 6. 메인 동작화면

메인 페이지에서는 인터넷 강의를 시청하면서, Pose estimation이 적용된 영상도 같이 출력한다. 거북목이나 기타 문제가 발생하면 영상에서 경고문을 출력하여 사용자가 알 수 있도록 한다. (그림 6)

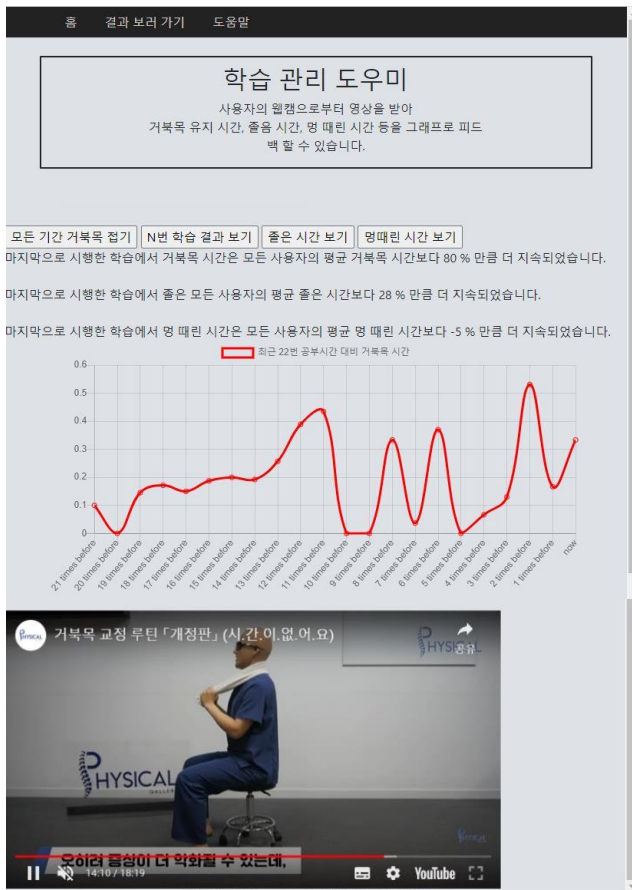


그림 7. 학습진단 및 추천 동영상 제공 화면

학습이 끝나면 자동으로 결과 창으로 연결되며, 결과 창에서는 학습 중에 얻은 데이터를 이용한 데이터 시각화로 피드백을 해주고, 데이터에 따른 추천 영상을 제공해준다. (그림 7)

3. 결과 고찰

위 Python 알고리즘을 단독으로 사용하면 Ⅲ-1.과 같은 일반 노트북 사양에서도 잘 동작한다. (그림 8)

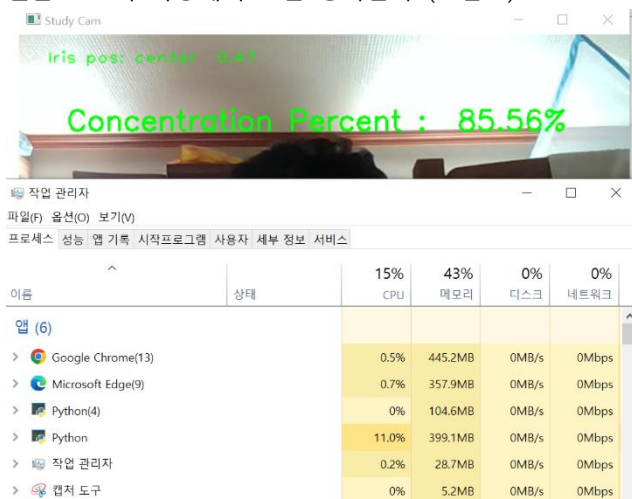


그림 8. Python 알고리즘 CPU 사용량

하지만 접근성을 고려하여 웹으로 동작하도록 설계하였다. Python 알고리즘을 웹으로 제공하는 데 있어 PyScript 를 사용하게 되었지만, PyScript 는 아직 알파단계이기 때문에 좋은 퍼포먼스를 기대하기 어렵다.

웹에서도 잘 작동하기 위한 방법은 PyScript 와 같은 프레임워크가 더 발전하거나, 사용한 알고리즘을 JavaScript 로 재작성 하는 방법이 있다. 이와 같은 방법이 적용된다면 더 좋은 퍼포먼스를 기대할 수 있다.

IV. 결론

본 논문에서는 Pose estimation 과 시선추적 기법을 적은 컴퓨터 자원으로, 접근성이 용이한 웹 애플리케이션으로써 제공하는 알고리즘을 제안하였다.

집중도라는 수치를 학습 중에 줄은 시간, 멍하게 있던 시간, 화면을 응시하지 않은 시간 등의 지표들을 이용하여 계산하였다. 하지만 언제나 합리적인 집중도 계산은 불가능하다. 집중을 하는데도 화면을 응시하지 않는 경우가 있기 때문이다. 예를 들어 강의 중 필기를 하는 상황이 있다. 이때 화면 응시를 하지 않았기 때문에 집중도 수치를 낮추는 현상이 발생할 수 있다. 그래도 위의 지표들을 제외하고 집중도를 판단할 지표는 무수히 많다. 본논문은 Pose estimation 과 시선추적 기법 사용에 국한되었기에 더 구체적인 계산은 불가능했지만, 다른 연구의 다른 기술을 추가한 집중도 분석에 이용될 것으로 기대한다.

참 고 문 헌

- [1] Francois Beletti, 2021, "MoveNet.SinglePose Model Card", Google
- [2] Ronny Votel(Google Research), "Next-Generation Pose Detection with MoveNet and TensorFlow.js", Tensorflow Blog, 2021.05.17
- [2] 정연우, 공원태, 권혁수, 2013, "머리전방자세에 따른 상체의 생체역학적 상관분석", 『대한정형외과학회지』, 2013;19(2)1-2
- [3] S. Y. Kim, N. S. Kim, and L. J. Kim, 2015, "Effects of cervical sustained natural apophyseal glide on forward head posture and respiratory function", J Phys Ther Sci, Vol.27, No.6, pp.1851-1854,
- [4] 김신용, 김세중, 손슬기, 동상옥, 이재철, 신동재. 2013. "두부 전방 정도와 경추 추간판 탈출증 발생 부위의 상관성 분석", 『척추신경추나의학회지』, 2013;8(2)33-35