

OpenCV 를 활용한 졸음인식 CNN 모델 제작

김주영, 김은혜, 전지은, 김명주*
 서울여자대학교 정보보호학과 학부생, 교수*
 pyosowe12061206@gmail.com
 rladms0103@gmail.com
 klara010149@gmail.com
 mjkim@swu.ac.kr

Development CNN Model of Drowsiness Detection Using OpenCV

Joo-young Kim, Eun-hae Kim, Ji-eun Jeon, Myuhng-Joo Kim
 Department of Information Security, Seoul Women's University

요 약

본 논문에서는 비대면 교육 상황이 확대되는 시점에서 자율 학습에 유용하게 사용할 수 있는 학습자의 졸음을 인식하여 알려주는 모델을 설계하여 구현하였다. 기계학습의 CNN 알고리즘을 활용하여 공부상태와 졸음상태를 판별하는 모델을 만들고, Opencv 을 사용하여 일정 횟수 이상 졸음상태가 반복되면 알람을 울려 사용자를 잠에서 깨운다. 이 프로그램은 자기 관리 및 독립적인 학습을 수행하는 데에 도움을 줄 수 있다.

1. 연구 배경

1.1. 연구의 목적

코로나 바이러스(COVID-19)의 영향으로 현재까지 많은 학교에서 비대면 수업을 진행하고 있다. 하지만 현장에서 직접 소통하지 않는 비대면 수업은 학생의 자제력과 의지력을 쉽게 떨어뜨릴 수 있다. 학생의 학습을 방해하는 요소로는 다양한 것이 있을 수 있지만 가장 큰 요인은 졸음이다. 다소 편안한 공간에서 수업을 듣고, 과제와 공부를 하면 스멀스멀 잠이 오기 마련이다. 슈터디(SWU-TUDY)는 학습 중에 조는 문제로 고민중인 학습자들을 위해 실시간으로 졸음을 인식해서 깨워줄 수 있는 모델을 만들고 구현하고자 한다.

1.2. 유사기술 동향 및 비교분석

졸음 인식 기술은 대표적으로 운전자 졸음운전 방지를 위해 쓰이고 있다. 지능형자동차 기술 중에 안전 시스템 분야에서 활발히 연구되고 있는 분야가 운전자의 졸음을 미리 감지하고 사고를 예방하는 사고 예방 보조 시스템이다. 운전자의 졸음 정도를 세 단계로 나뉘 졸음 정도가 낮으면 알람을 울리고, 중간일 때는 전동 시스벨트를 연동시켜 경보를 전달하고, 최고 수준이면 차량 안전제어 시스템에서 제동을 할

수 있는 신호를 보내주는 식이다[1]. 졸음 운전 감지 및 방지 시스템은 현대모비스, 일본 토요타사에서 개발한 렉서스, 미국 포드 승용차 등에서 적용된 사례가 있다[2].

졸음 인식 기준으로 Fig. 1 과 같이 하품, 눈, 고개 숙임 등이 있다[3]. 먼저, 평상시 입의 너비에 대한 높이 비를 기준으로 2 배 이상의 비가 10 프레임 이상 연속으로 측정되었을 때 하품이라고 인식한다. 눈으로 졸음을 인식하는 경우 눈동자의 높이가 평상시 눈의 50% 미만이고 20% 이상일 경우 졸린 상태로 인식하며, 평상시 눈의 20% 미만일 경우 눈이 감겨 잠이 든 상태로 인식한다. 따라서 눈동자의 높이가 평상시 눈의 50% 미만인 상태로 0.4 초 이상 유지할 때 졸음으로 판단한다. 다음으로, 평상시 코끝과 턱의 거리에 대한 코의 시작과 코끝의 거리의 비를 기준으로 1.8 배 이상의 비가 측정되었을 때, 고개 숙임으로 인식한다.

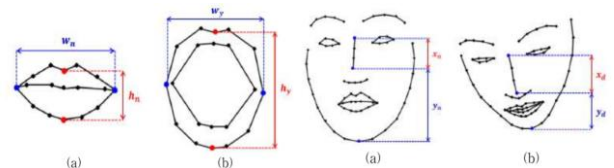


Fig. 1. Length of Mouth and head

1.3. 본 연구의 차별성

기존의 졸음 인식은 보통 운전자의 안전 운전을 위해서 연구되었다. 하지만 온라인 강의가 비대하게 증가한 오늘날, 졸음 인식은 학습자에게도 필요한 기술이다. 슈터디는 학습 도중 학생의 졸음을 인식하고, 깨워주는 조력자의 역할을 한다.

2. 시스템의 설계 및 구현

2.1. 시스템 동작 개요

슈터디는 CNN 모델을 제작하고, 이를 Opencv 에 적용하여 연결된 웹캠으로 사용자의 얼굴을 인식한다. 공부 중 사용자의 졸음이 인식이 되면 졸음 count 가 쌓인다. 졸음을 카운트하는 변수가 100 회 누적이 되면 시끄러운 알람이 울려 사용자의 잠을 깨워준다.

2.2. CNN 모델 설계 및 라이브러리 선정

두 가지 상태를 구분 짓는 모델을 만들기 위해 이미지를 기반으로 학습과 분류를 하는 CNN 기법을 채택하였다. 정확한 인식을 위해 얼굴에 랜드마크를 찍는 방식을 사용하여 얼굴과 눈동자의 움직임을 세세하게 인식하는 방법도 고려했으나, 웹캠의 화질로는 공부할 때 책을 내려보는 모습과 눈을 감은 모습이 동일하게 보였다. 따라서 다수의 이미지를 데이터셋으로 구성하고 학습시키는 방법인 CNN 이 노트북을 이용한 얼굴인식에 가장 적합하다고 판단했다. CNN 은 이미지를 인식하는 대표적인 딥러닝 모델이다. 이미지가 입력 되면 필터를 통해 특징을 추출한 뒤 그것을 여러 특징으로 분류하여 파악한다. 목표한 프로젝트를 구현하기 위해 텐서플로(Tensorflow) 라이브러리와 코랩(Colab) 플랫폼을 선택해서 개발을 진행하였다.



Fig. 2. Model design process

2.3. CNN 알고리즘 설계

프로젝트에서는 2 개의 컨볼루션(Convolutional) 계층, 2 개의 맥스풀링(Maxpooling) 계층, 3 개의 드롭아웃(Dropout) 계층, 2 개의 덴스(dense) 계층, 1 개의 플래튼(Flatten) 계층을 사용하였다. 은닉층에서는 렐루(relu) 활성화함수를 사용하였다. 첫 번째 컨볼루션 계층에서 필터의 개수는 16 개, 필터의 사이즈는 3x3 으로 정의하였다. 컨볼루션 에서는 필터의 개수만큼 특성 맵이 나오는데, 복잡한 출력을 작게 만들어서 불필요한 정보를 간추리기 위해 맥스풀링층을 쌓아 특성 맵의 크기를 줄였다. 맥스풀링 계층의 풀링 창 크기는 2 로 설정하여 전체크기를 절반으로 줄였다. 그리고 과적합을 예방하기 위해 드롭아웃 계층을 쌓아 은닉층에 배치 된 노드 중 25 퍼센트를 꺼서 학습데이터

에 지나치게 치우쳐져 학습 되는 현상을 방지하였다. 그리고 또 한번 컨볼루션 계층을 쌓았다. 이때 필터의 개수는 64 개, 필터의 사이즈는 3x3 이다. 이어서 풀링 창의 크기가 1 인 맥스풀링 계층을 쌓고, 25 퍼센트의 노드를 끄는 드롭아웃 계층을 쌓았다. 그리고 컨볼루션과 맥스풀링은 이미지를 2 차원 배열로 다루기 때문에 이를 1 차원으로 변환하는 플래튼 함수를 사용하였다. 덴스 함수를 통해 256 개의 노드를 주고, 드롭아웃으로 50 퍼센트의 노드를 깎았다. 마지막 덴스 출력층에서는 소프트맥스(softmax) 활성화함수를 통해 총합이 1 인 형태로 변환시켰다. 큰 값이 두드러지게 나타나고, 작은 값은 더 작아지도록 하기 위함이다. 오차함수는 이항 교차 엔트로피(binary_crossentropy)를 사용하여 0 과 1 두 가지 결과로 나오게 하였고 최적화기는 아담(Adam)을 사용하였다[4].

2.4. 학습 데이터 준비

첫 과정으로는 졸음을 인식하는 데에 필요한 학습 데이터셋을 만드는 것이다. 먼저 클래스를 졸음상태, 공부상태 두 가지로 나누어 데이터셋을 구성하였다. 데이터셋은 각 클래스 별로 750 장 가량이 되며 jpg 형식으로 설정하였다. 데이터셋은 팀원들이 직접 사진을 촬영하여 구성한 이미지들이다. 졸음상태 클래스에는 흔히 앉아서 졸 때의 모습들을 넣었다. 고개가 고꾸라져서 조는 모습, 아예 엎드린 모습, 카메라 상에서 아예 벗어난 모습들이 있다. 공부상태 클래스에는 책을 보는 모습, 턱을 괴고 책을 보는 모습, 노트북에서 멀리 떨어져서 책을 보는 모습, 노트북을 쳐다 보는 모습, 화면 밖으로 조금 나간 상태로 공부하는 모습 등 앞에 놓인 노트북을 기준으로 가능한 자세의 모든 각도에 대비하여 데이터셋을 구성하였다. 모델을 만들 때 이미지 사이즈는 28x28 로 조정하였고, opencv 에서도 실행환경 사이즈를 동일하게 조정하여 호환이 되게 하였다. 또한 최적의 성능을 내기 위하여 이미지 데이터를 0 에서 1 사이의 값으로 변환하는 데이터 정규화를 실시하였다

2.5. 모델 학습 과정

model.fit()을 통해 정해진 환경을 주어진 데이터를 불러 실행시켰다. 이때 EarlyStopping()함수를 이용하여 과적합을 방지하는 코드를 추가하였다. 학습이 많이 진행되면 정확도는 올라가지만 과적합 문제로 오히려 실험 결과가 나빠질 수 있기 때문이다. 그래서 학습이 진행되어도 테스트셋의 오차가 줄지 않으면 학습을 멈추게 하였다[5]. 학습에서는 에포크(epochs)를 3500, 배치사이즈(batch size)를 10 으로 설정하여 각 샘플을 처음부터 끝까지 3500 번 재사용 될 때까지 실행을 반복하고, 전체 이미지 샘플을 10 개씩 끊어서 집어넣었다.

2.6. 모델 테스트

제작이 완료 된 모델이 이미지를 제대로 학습했는지 테스트를 하는 과정을 거쳤다. 졸음 상태 혹은 공부 상태의 이미지를 동일한 환경으로 조정 및 정규화

를 하고, 이 이미지를 모델에 로드하여 판별되는 클래스를 출력하게 하였다. 이때 이미지들은 학습 시 사용되었던 데이터 셋의 일부이다. 결과는 95 퍼센트로 올바른 클래스를 출력하였다. 학습할 때 넣었던 이미지들은 모델에 제대로 학습이 된 것을 확인하였다.

2.7. opencv 모델 적용

신경망의 입력 데이터에 특정 변환을 가하는 전처리 작업을 실행했다. 모델에 의존성을 가지므로 28x28 이미지인 학습사이즈와 같은 사이즈로 행렬의 행 크기를 변환한다[6]. 또한 각 데이터들을 해당 데이터 셋이 가질 수 있는 값의 범위로 나누어 정규화를 하였다. 현재 CNN 입력은 정규화 된 4 차원 데이터가 필요하다. 따라서 예측을 위해 데이터 형태(1, height, width, channel)의 형식으로 행렬의 모양을 변형시키는 reshape()함수로 이미지의 크기를 전부 통일시켰다[7]. 또한 공부 판별, 졸음 판별, 알람 기능을 수행하기 위해 함수와 반복문으로 기능을 구현하였다. 윈도우 내장 알람은 winsound 를 import 하여 구현하였다[8]. 판별 방식은 웹캠으로 들어오는 사용자의 이미지들을 모델을 통해 예측률을 파악한다. 공부클래스에 더 가까운 예측을 한다면 공부상태로, 졸음클래스에 더 가까운 예측을 한다면 졸음상태로 판별한다.

3. 실행 결과 분석

3.1. 개발 최종 결과물

졸음인식 프로그램은 텐서플로를 통해 만든 졸음인식 모델을 opencv 로 실행시키면 사용자의 얼굴 움직임을 실시간으로 탐지한다. 사용자의 졸음이 100 회 이상 카운트 되면 시끄러운 알람으로 깨워주는 시스템이다. 비대면 시대에는 주로 앞에 노트북을 두고 공부하기 때문에 자기 관리 및 졸음 관리가 부진했던 이용자가 도움을 받을 수 있다. 그리고 지속적으로 사용자의 상태를 인식하기 때문에 사용자는 관리 및 감시를 받고 있는 느낌을 받아 집중도를 높일 수 있다. 또한 실행시간이 짧고 사용법이 매우 간편하여 시간을 허비하는 일 없이 공부를 시작할 수 있다. 출력은 사용자가 간결하게 원하는 상태 정보만 알 수 있도록 하였다.

3.2. 공부상태 인식

프로그램은 특정한 공부 자세에 국한하지 않고 다양한 자세를 공부상태로 인식한다. 공부하면서 자세로만 있는 사용자는 드물기 때문에 그에 대비하여 여러 각도와 자세를 공부상태로 인식할 수 있게 데이터 셋을 구성했기 때문이다. 턱을 편 자세, 가까이 화면을 보는 자세, 멀리 화면을 보는 자세, 비스듬하게 화면을 보는 자세, 책을 내리 보는 자세, 고개를 좀 더 숙여서 책을 보는 자세 모두를 공부상태로 인식한다.

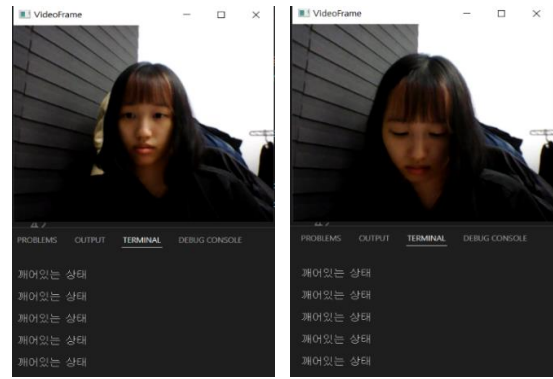


Fig. 3. Study state Recognition

3.3. 졸음상태 인식 및 알람 송출

이용자가 졸기 시작하여 고개가 연속적으로 고꾸라지고 정수리가 보이면 졸음상태로 인식한다. 졸음을 카운트하는 변수가 100 회가 되면 윈도우 내장 알람음이 크게 울려 사용자를 잠에서 깨운다. 졸음변수가 누적되는 중 다시 공부상태로 인식 된다고 해서 초기화하지 않고 계속 누적이 된다. 조는 과정에서 고개를 잠시 똑바로 드는 찰나의 모습이 공부상태로 인식될 수 있는데, 변수를 초기화 하지 않아야 조는 중에 알람을 울릴 수 있기 때문이다. 알람이 한번 울렸다면 졸음 변수는 0 으로 초기화되고 다시 100 회가 되면 알람을 울리는 방식으로 작동한다.

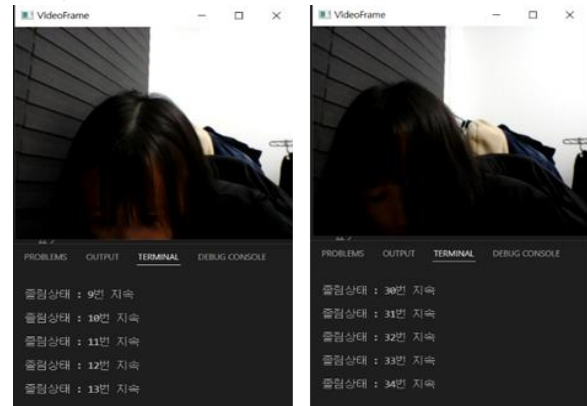


Fig.4. Sleep-detection Recognition

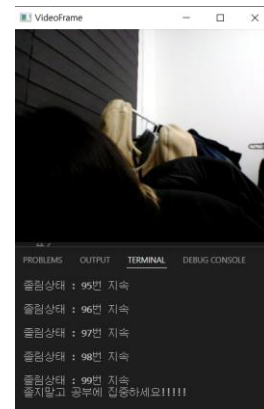


Fig. 5. Alarm and message output when 100 times of sleepiness are accumulated

3.4. 얼굴인식 정확도 파악

정확도를 파악하기 위하여 prediction 배열을 선언하고 이에 model.predict() 값을 대입하였다. 코드에서는 클래스 예측의 근거로 prediction 배열값을 사용한다. prediction[0,0]은 클래스 1 즉 공부상태 클래스에 대한 정확도를 담고 있고, prediction[0,1]은 클래스 2 인 졸음상태 클래스에 대한 정확도를 가지고 있다. prediction[0,0] > prediction[0,1] 이라면 모습이 클래스 1 에 더 가까운 것이므로 공부상태로 출력한다. 반대의 경우에는 졸음상태로 출력한다. 현재 구현 된 모델이 얼마나 상태를 정확하게 인식하는지 알아보기 위해 prediction 값을 같이 출력해보았다. 입력 된 CNN 알고리즘은 현재 인식되는 화면이 확실하게 한쪽 클래스에 부합한다면 1.0 의 값을 출력한다. 1.0 은 퍼센트로 표현했을 때 100 의 의미이다. 양쪽 클래스 데이터셋과 겹치는 모습이려면 1.0 미만의 예측도를 출력하고 더 큰 확률을 가진 클래스를 출력한다.

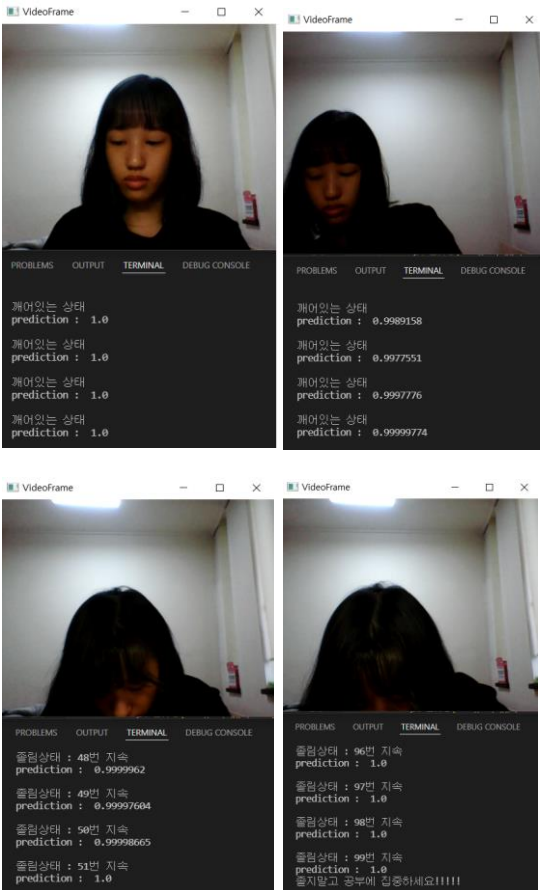


Fig.6. Accuracy of CNN model

3.5. 성능 개선안

현재 구축한 인공지능 모델을 웹에 올리는 과정에서 애쓰고 있다. 언택트 시대에 맞춰 공부하는 방식도 발 빠르게 변화하였다. 유튜브, 줌, 행아웃, 구루미 등 온라인 상에서 본인이 공부하는 모습을 촬영해 올리고, 카메라를 켜서 다른 사람들과 함께 공부하는 문화가 확산되었다. 흔히 스터디윗미(Study with me)라고도 불린다. 슈터디는 학습자가 스터디윗미 사이트에 접속해 웹캠을 켜 상태에서 공부를 하다가 즐기

시작하면 졸음을 인식하여 관리해주는 공부관리 웹 서비스를 개발할 예정이다. 지금까지는 목표 관리, 일정 관리, 성적 통계를 측정하는 기능을 구현하였다.



Fig. 7. Schedule & Grade point page

앞으로 졸음인식이 일정횟수를 초과하면 부모님이나 친구 등 주변인에게 문자를 발송하여 현재 공부중인 상황을 실시간으로 알리는 기능을 추가할 것이다. 카메라의 해상도를 높일 수 있다면 모바일 앱에도 적용하여 휴대폰으로 간편하게 사용할 수 있도록 한다. 또한, 졸음 인식을 넘어 집중도를 측정하는 모델로도 개선할 수 있을 것이다.

3.6. 기대효과 및 활용분야

기존 일정과 목표만을 관리하는 스터디 웹과는 다르게 슈터디는 사용자의 졸음을 인식 및 기록하여 학습자가 강의에 집중할 수 있게 한다. 피곤해서 강의에 집중하지 못하는 학생들에게 슈터디는 실시간으로 학생들 결을 지키며 졸음을 깨우는데 큰 도움이 될 것이다. 나아가 다수의 학생들을 비대면으로 관리 감독하는 것이 쉽지 않은데, 알람이 뜨는 학생들을 위주로 교육자가 주의 깊게 살펴볼 수 있어 학생들만이 아닌 교육자의 입장에서도 다방면으로 활용 높게 이용될 수 있다.

참고문헌

- [1] 최진모, 송혁, 박상현, 이철동 “운전자 졸음 인식 시스템 구현” 한국통신학회논문 v.37C no.8 pp.711-720 (2012)
- [2] 안병태 “졸음운전 감지 및 방지 시스템 연구” 융합정보논문지 v.8 no.3 pp.193-198 (2018)
- [3] 오미연, 정유수, 박길흠 “얼굴 특징점 기반의 졸음 운전 감지 알고리즘” 멀티미디어학회논문지 v.19 no.11 pp.1852-1861 (2016)
- [4] 조태호 “모두의 딥러닝” 길벗 pp.122-129, pp.204-233 (2017)
- [5] 사이토고키 “밑바닥부터 시작하는 딥러닝” 한빛미디어 pp.96-102 (2017)
- [6] 오준석,이진영,임태원,옥승호 “얼굴인식 및 눈 깜빡임을 사용한 비접촉식 보안시스템” 추계학술대회.대한전자공학회.2021
- [7] n.d “opencv-python study documentation!” 2022 년 1 월 26 일접속 <https://opencv-python.readthedocs.io/en/latest/>
- [8] 김효실 “파이썬 생활밀착형 프로젝트” 로드북 pp.120-144 (2021)