

비지도 학습 기반 눈 깜빡임 이상치 탐지*

이준범^o 박상근

경희대학교 소프트웨어융합학과

jun2mun@naver.com, sk.park@khu.ac.kr

Unsupervised Training Based Eye Blinking Outliers Detection

Junbeom Lee^o Sangkuen Park

Department of Software Convergence, Kyunghee University

요 약

본 논문에서는 LSTM-AE(LSTM AutoEncoder) 모델을 기반으로 눈 깜빡임 데이터 분포의 이상치를 감지한다. 대부분의 기존 연구는 눈 깜빡임 데이터가 기준치에 미달하거나 초과하는 경우 이상치로 분류하였다. 이는 이상치 기준을 설정하는 데에 있어 불특정 다수의 눈 깜빡임 패턴을 정상 데이터로 사용하여 사용자별 데이터를 반영하지 못한다. 본 연구에서는 30fps의 웹캠을 통해 측정되는 연속적인 EAR(eye aspect ratio) 데이터에서 LSTM-AE(LSTM AutoEncoder) 모델을 기반으로 단위 시간당 눈 깜빡임 이상치를 검출하는 방법을 제시하였으며, 모델 평가 결과, ROC를 기준 0.958, 정확도 92%로 이상치를 검출할 수 있음을 확인하였다.

1. 서 론

최근 코로나 시대를 지나면서 전자기기 사용 시간이 증가하였다. 2021년 기준으로 스마트 기기 사용 시간은 2016년 대비 44%가량으로 증가하였으며, 전체적으로 증가하는 추세에 있다 [1]. 컴퓨터 비전 증후군이라고도 알려진 디지털 눈의 피로(DES)는 다양한 안구 및 시각 증상을 포함한다. DES는 눈 깜빡임 반사와 눈 안쪽 근육을 조인다는 문제가 있으며 [3, 4], 컴퓨터 사용자들 사이에서 DES의 유병률이 50% 이상일 수도 있다고 한다 [2]. 최근 연구는 전 세계의 70%의 컴퓨터 사용자가 시력 문제를 겪고 있음을 보인다. 더 문제인 것은 이에 영향을 받는 사람들이 꾸준히 증가하고 있다는 것이다. [5] DES 증상이 반복될 경우, 시력 저하, 난시 등과 같은 교정되지 않은 시력 문제 노안 등의 문제를 야기한다. 하지만, 일에 집중하고 있을 때, 적절한 휴식 시간을 갖는 것은 어렵다 [6]. 일부 소프트웨어는 사용자들이 사전에 지정된 시간 경과 시 휴식 시간을 가질 수 있도록 관리를 돕는다. 사용자의 컴퓨터 사용 시간과 마우스 사용 시간 등의 데이터를 통합하여 기준치 초과 시, 팝업을 통해 사용자에게 휴식 알림을 주는 애플리케이션이 있다 [7,8]. 하지만 사용자가 현재 본인의 눈 피로도 상태를 인지할 수 있는 기능을 제공하지는 않았다. 눈의 깜빡임 시간과 빈도수를 통해 눈의 피로도를 감지하는 연구도 있다 [6]. 하지만 이는 불특정 다수가 오랜 시간 컴퓨터를 사용한 표본을 활용하여 눈 피로 여부를 라벨링 하였기에 개인의 눈 깜빡임 패턴과 동일하다고 일반화하기 어려울 수 있다. 따라서 본 연구는 개인의 눈 깜빡임 분포를 기반으로 이상치가 발생하였을 때 사용자에게 알림을 줄 수 있는 눈 깜빡임 이상치 탐지 모델을 제안한다.

* 본 연구는 과학기술정보통신부 및 정보통신기획평가원의 2023년도 SW중심대학사업의 결과로 수행되었음 (2023-0-00042).

2. 관련 연구

이상치 탐지 연구 방식은 다양한 방식으로 진행된다. LSTM-AE(LSTM Auto Encoder) [17] 방식의 이상치 탐지는 입력을 그대로 출력 해내도록 한다. 이는 “눈 깜빡임” 데이터 타입과 같은 “네트워크” 데이터 분석에서도 활용된다 [14]. 이는 시계열 데이터에서 이상치를 감지하는 연구에서 높은 성능이 보임을 알 수 있다. 또한 Isolation Forest 방식을 통해서 눈 깜빡임의 행위를 이상치로 감지하는 방법 또한 진행된다 [15]. 그리고 IQR(Inter Quartile Range)를 활용하여 이상치를 제거하는 연구 또한 진행되고 있으며 높은 성능을 보였다 [16].

본 연구는 IQR과 LSTM-AE 모델을 활용하여 이상치를 감지한다. 사용자별 EAR 분포 데이터를 통해, 분포에 이상치가 생길 경우 이를 감지하고 사용자에게 알림을 줄 수 있다. 가장 기본적인 이상치 기준 설정 방법은 IQR을 사용하는 방법이다. 하지만 IQR 방식만을 활용할 경우, 점점 피로해지는 경우의 이상치 감지가 어려우며, 새로운 데이터가 계속 들어오는 경우에는 전체 데이터의 분포를 표현하는 데에 한계가 있다. 본 연구는 LSTM-AE 모델과 IQR을 함께 활용해서 학습된 기존 눈 깜빡임 분포와 어긋난 눈 깜빡임 이상치를 검출해 내는 방법을 제안한다.

2.2 눈 깜빡임 연구

인간의 눈 깜빡임에는 비자발적 눈 깜빡임, 반사적 눈 깜빡임, 자발적 눈 깜빡임의 세 가지 유형이 있다. 그리고 안구건조증 환자는 눈물 증발 속도가 빨라지고 비자발적으로 깜빡이는 횟수가 증가하는 것으로 보고 되었다 [10]. 또한 스마트 기기 사용 중 자발적 눈 깜빡임 횟수(SBR)의 중앙값이 더 낮은 경향이 보임을 확인하였다 [11]. 또한 눈 깜빡임 빈도수(BF), 평균 눈 깜빡임 경과시간(ABD), 눈 감는 소요 시간(AECD)의 감소가 눈 깜빡임 반사의 감소와 상관관계가 있음을 연구하였다 [6]. 또한

불완전한 깜빡임이 눈의 피로도에 영향을 미치며 실제 안구건조증 환자는 눈을 깜빡이는 속도가 느리며 불완전한 횡수가 높다는 것으로 결론 내었다[10]. 이러한 수치에 어긋나는 패턴, 즉 눈을 지나치게 깜박이지 않거나 EAR 값의 변화가 일정 시간 없을 경우 그리고 눈 깜빡임 간격이 비정상적일 경우 이상치로 판단하는 연구가 이루어지고 있다[12]. 본 연구에서는 눈의 비정상 상태인 경우, 눈 깜빡임 횡수가 낮아진다는 연구 자료를 활용한다. 또한 불완전한 깜빡임으로 인해 눈 깜빡임의 횡수가 비정상적으로 변화한다는 것을 연구에 활용한다.

3. 설계 및 구현

3.1 모델 선정

눈 감지 모델은 Google의 mediapipe를 활용한다. 그중 Face mesh 모델을 통해 얼굴에 랜드마크를 한다. Face mesh 모델은 478개의 3차원 지점을 제공한다 [13].

눈 깜빡임 데이터의 수집은 EAR(eye aspect ratio) 모델을 통해 수집한다. EAR은 눈 주위의 여섯 개의 지점 (p) 간 유클리드 거리(Euclidean distance)를 통해 계산된 값이다. 이는 눈을 뜨고 감는 동안 변화하는 값이다 [9].

$$EAR = \frac{\|p_2 - p_6\| + \|p_3 - p_5\|}{2\|p_1 - p_4\|}$$

이를 활용하여 눈이 일정한 EAR 이하로 감소할 경우 눈 깜빡임을 감지할 수 있으며, 눈 감김 패턴을 측정하는 데 용이하게 활용된다.

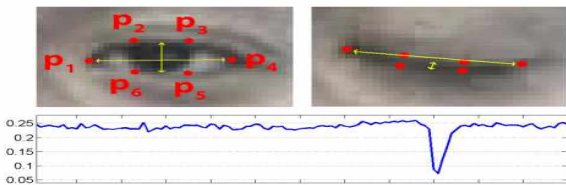


그림 1 프레임별 EAR 계산 [9]

기존 연구에서는 EAR 데이터를 통계 자료에 따라 정상/비정상 라벨링을 한다. 하지만 사용자별로 정상/비정상의 기준은 다를 수 있다. 따라서 사용자의 눈 깜빡임 분포의 이상치를 전부 반영하기에 어려움이 있다.

눈 깜빡임 데이터의 특징은 시계열 데이터라는 점이다. 기존의 EAR 데이터에서 비정상 눈 깜빡임 데이터는 정상적인 자발적 눈 깜빡임 데이터 보다 적다고 볼 수 있다. 따라서 LSTM-AE 모델을 활용하기에 적합하다. 학습을 진행하기 전, IQR 분포를 기준으로 노이즈 데이터를 제거하고 데이터 정규화를 진행한다. 정규화 방식으로는 이상치 문제를 줄이기 위하여 Z-score 정규화 방식을 채택하였다.

3.2 실험 데이터

눈 깜빡임이라는 목적에 맞는 데이터 셋을 활용하기 위해 다음 조건에 맞는 데이터를 수집한다. 동일 수준의 각도, 조도, fps를 갖는 데이터 수집 환경으로 설정한다. 데이터 수량은 인당 10,000초 이상의 데이터를 수집한다. 얼굴의 허용 각도는 정면을 기준으로 상하좌우 약 30도 이내로 설정한다. 조도의 경우 지나치게 어두운 환경이나 밝은 환경, 실외의 환경을 제외한다. fps의 경우 30프레임 내외로 설정한다.

이에 적합한 AI HUB의 원시 영상 데이터셋*을 활용하였다. 추가로 동일한 조건 내에서 웹캠을 통해 실험자 3명의 데이터를 수집하였다. 동일하게 1초 간격으로 측정된 눈 깜빡임 횡수 10,000개를 활용하였다. 이를 Window Size 60의 시계열 데이터로 전처리 하였다. LSTM-AE 모델 학습을 진행하기 위하여, 먼저 IQR 값을 기준으로 75%보다 크거나 25%보다 작은 구간의 데이터를 이상치로 판단하여 제거했다. 정상 데이터의 훈련(train) 데이터는 80%로 테스트(test) 데이터는 20%로 설정하였다. IQR 분포에서 이상치로 판단한 데이터의 30%는 테스트 데이터와 랜덤으로 섞어 평가한다. 최종 테스트 데이터는 1000개, 검증(validation) 데이터는 500개로 구성되었다. 훈련을 통해 정상/비정상을 구분한 임계값을 지정하기 위해 Precision Recall Curve를 적용하였다. 이를 통해 재구성 손실(Reconstruction Error)을 구하고, AUC Curve로 모델을 평가했다.

3.3 실험 결과

학습은 Epoch 200회를 기준으로 진행하였다. 학습 시 과적합을 방지하기 위하여 조기 종료를 도입하였다. 10회 가량 검증 데이터의 loss가 증가할 경우, 종료하도록 설정하였다.

표 1 실험 결과 표

	ROC	Accuracy
Case1	0.958	92%
Case2	0.978	93%
Case3	0.973	94%

표 1의 case 1의 경우, AI HUB 데이터를 활용하여 훈련한 결과를 평가한 것이다. case 2와 3의 경우 웹캠을 통해 수집한 데이터를 훈련한 결과를 평가한 것이다. 실험 결과는 [표 1]과 같다. 92~94%가량의 정확도로 이상치를 감지하는 것을 알 수 있다. 또한 ROC 평가에서도 0.95 이상을 보이며 높은 평가를 받았다.

* AI-HUB [딥페이스 변조 영상 데이터셋 중 원시 데이터 활용]

3.4 어플리케이션 설계 결과

본 연구에서는 이상치가 발생하였을 때, 사용자에게 즉시 알림을 주어 눈 깜빡임을 의식할 수 있도록 한다. 그림 2 (1)은 현재 사용자의 눈 깜빡임 상태를 모니터링 할 웹캠의 활성화 여부이다. 활성화되어 있을 경우, 그림과 같은 상태가 된다. 사용자의 데이터가 입력된다면, 그림 2 (2)와 같이 EAR 값 분포를 모니터링할 수 있다. 이를 통해 사용자는 본인의 눈 깜빡임 상태를 확인할 수 있다. 또한 그림 2 (3) 스크린 타임 기능 제공을 통해 프로세스 사용 시간 순위를 알 수 있다. 그리고 그림 2 (4)와 같이 EAR 분포에 이상치가 감지 시, 사용자에게 알림을 주는 기능을 제공하며 EAR 분포가 정상으로 돌아올 경우, 알림을 제거한다. 이를 통해 일에 집중하고 있을 때, 적절한 휴식 시간을 갖지 못하는 문제를 해결하고자 한다 [6].

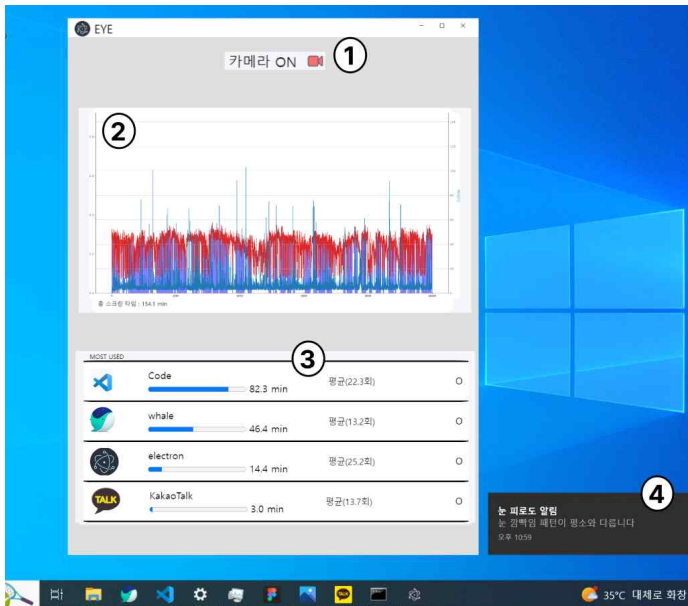


그림 2 어플리케이션 구동

4. 결론 및 향후 연구

본 논문에서는 EAR 모델을 활용하여 눈 깜빡임을 추출하였다. 이 깜빡임 데이터에서 이상치를 감지하였다. 이상치 검출 방식 중 LSTM-AE 방법을 활용하여 높은 성능의 ROC를 결과로 얻을 수 있었다. 이를 활용해 어플리케이션으로 알림 기능을 더하였다. 그러나 데이터가 다양한 연령대, 측정 시각에 따라 달라질 수 있다는 점이 있기에 추가 데이터를 통한 검증이 필요하다. 따라서 추가 실험자와 더 높은 fps의 장비를 활용하여 학습을 한다면, 정확도를 더 개선할 여지도 있음을 알 수 있다.

5. 참고 문헌

[1] 문화체육관광부. 2021 「국민여가활동조사」 표 1-27.
 [2] Sheppard, Amy L., and James S. Wolffsohn. "Digital eye strain: prevalence, measurement and amelioration." BMJ open ophthalmology 3.1 . 2018.
 [3] J.Bali, N. Narvin, B. R. Thakur: &Computer vision

syn-drome: A study of the knowledge, attitudes and practices in Indian Ophthalmologists&, &Indiian Journal of Ophthalmology&. 2007.

[4] "Working with VDUs", Health and Safety Executive Books, UK, December 2006.

[5] Blehm, Clayton, et al. "Computer vision syndrome: a review." Survey of ophthalmologyvol 50.3 p .253-262. 2005.

[6] Divjak, Matjaz, and Horst Bischof. "Eye Blink Based Fatigue Detection for Prevention of Computer Vision Syndrome." MVA. p. 350-353. 2009.

[7] workpace, wellnomics Ltd, <http://www.workpace.com>.

[8] EyeShield, Nimble software.

<http://www.nimblesoftware.com/desktop/eyeshield>

[9] Soukupova, Tereza, and Jan Cech. "Eye blink detection using facial landmarks." 21st computer vision winter workshop, Rimske Toplice, Slovenia. p.2 . 2016.

[10] Su, Yuandong, et al. "Spontaneous eye blink patterns in dry eye: clinical correlations." Investigative ophthalmology & visual sciencevol. 59.12 p. 5149-5156. 2018.

[11] Kuwahara, Akihiro, et al. "Eye fatigue estimation using blink detection based on Eye Aspect Ratio Mapping (EARM)." Cognitive Robotics2 p. 50-59. 2022.

[12] T. Jung, S. Kim and K. Kim. "DeepVision: Deepfakes Detection Using Human Eye Blinking Pattern," in IEEE Access. 2020.

[13] Lugaresi, Camillo, et al. "Mediapipe: A framework for building perception pipelines." arXiv preprint arXiv:1906.08172. 2019.

[14] Said Elsayed, Mahmoud, et al. "Network anomaly detection using LSTM based autoencoder." Proceedings of the 16th ACM Symposium on QoS and Security for Wireless and Mobile Networks. p.37-45. 2020.

[15] Liu, Fei Tony, Kai Ming Ting, and Zhi-Hua Zhou. "Isolation forest." 2008 eighth ieee international conference on data mining. IEEE, 2008.

[16] Pontifex, Matthew B., et al. "Variability of ICA decomposition may impact EEG signals when used to remove eye blink artifacts." Psychophysiology54.3 p. 386-398. 2017.

[17] Srivastava, Nitish, Elman Mansimov, and Ruslan Salakhudinov. "Unsupervised learning of video representations using lstms." International conference on machine learning. PMLR, p.843-852. 2015.