Novel machine learning pipeline for real-time oculometry.

Albert Sledzianowski[1], Jerzy P. Nowacki[1], Konrad Sitarz[1], and Andrzej W. Przybyszewski[1]

The Faculty of Information Technology, Polish-Japanese Academy of Information Technology, Warsaw, 02-008, Poland.

Abstract. In modern medicine, there is a need for widely available and efficient oculometric systems. This is because oculometry provides a simple insight into the general motor status of the patient. A good example are neurodegenerative diseases (ND) such as Parkinson's disease (PD) with bradykinesia as one of the effects causing a general slowing of movements, also visible in eye movements. PD develops secretly for many years before showing visible effects, but eye -tracking tests can reveal it sooner. That is why we decided to create a modern system for performing this type of tests based on neural network models, modern inference approach and algorithms of various types to help obtain reliable results. Our idea was to disconnect the software from the hardware, and base the hardware requirements on standard household equipment, such as a standard computer camera. This step is necessary because modern telemedicine must necessarily be based on basic equipment available to patients in their households. Our results showed that even with 30 Hz frequency and a standard reflexive-saccade (RS) test, with our system we were able to distinguish the results of healthy person from PD patients or old person. This solution brings the ability to receive quantitative online data, in opposition to standard telemedicine allowing only for verbal and/or visual interactions with the patient. We hope that the low hardware requirements will popularize automated oculometric tests in the context of age-related complications and ND diseases, which may contribute to earlier diagnoses and the collection of the appropriate amount of data to develop new means of preventing different diseases.

Keywords: machine learning, deep learning, eye moves, eye tracking, oculometry, reflexive saccades, saccade detections, neurodegenerative diseases, Parkinson's disease

1 Introduction

The video-oculography is the non-invasive, video-based method of eye tracking allowing for measuring all components of the eye movements. which gives hope for the creation of publicly available systems in the future to support the detection of ND and measure its progress. Many researchers recently showed

very good results of video-oculography and oculometric tests in classification or prediction of different ND,

Semmelmann et al showed good example of JavaScript algorithms for eye tracking with standard webcams to to study fixation, pursuit, and free viewing with to analyze the spatial precision in the first two, and repeatability of well-known gazing patterns in the third task [1]. The main focus of the study was to test the potential and possible limitations of webcam in terms of noise and accuracy to find differences between laboratory-grade equipment and webcam results [1]. They concluded that web-based eye tracking is suitable to conduct all three tests with little less accuracy and higher variance of the data [1].

Lin et al. developed a real-time eye-tracking system based on a webcam and machine learning models, combining a position criterion, gaze direction detection and lightning filtering [2]. The gaze direction estimation was based on appearance-features of eyes, Fourier Descriptor and the Support Vector Machine [2]. Authors concluded that their webcam solution brought quality results with feasibility of the solution [2].

Meng et al investigated Convolutional Neural Network models (CNN) with webcams with very promising results for this type of eye-tracking method by detecting 6 eye features to gather more exact eye movement data [3].

Aljaafreh et al [4] created an oculometric system based on webcam and application projecting dynamic stimulus. It was tested in research on 10 Multiple Sclerosis (MS) patients. The system calculated multiple parameters of saccades and both latency and amplitude gave sufficient results that allowed for diagnosis for MS disorder making this system sensible for clinical diagnosis and identification of neurological disorders [4].

The RS are a very good example of coulometric tests that give good results in the diagnosis of ND. The RS are rapid, ballistic eye movement of the eyes triggered by the influence of an emerging stimulus, with the purpose to direct the gaze focus to this interesting element of the environment. Because different regions of the brain are involved in this process, therefore, RS are very sensitive to degenerative changes caused by ND and also by human aging. The analysis of the RS parameters, mainly the latency showed in many cases correlation with development of different ND, i.e. PD, Alzheimer's Disease (AD), Frontotemporal Dementia (FA) or Amnestic Mild Cognitive Impairment (MCI) [5–8, 8–17].

Our idea was to build an advanced pipeline that would allow for eye-tracking based on CNN and video stream from a webcam, while controlling the information about the surrounding conditions, because a webcam can provide poor quality and frequency, and hardly depends on the lighting environment. Additionally we equipped our system with the ability to filter and smooth the signal to balance fluctuations, as well as the ability to detect saccades in the movements of the eye and calculate its parameters in an acceptable error range. Therefore, in addition to surroundings validation and signal smoothing, the saccade detection algorithms are oriented towards a lower sampling rate, with assumption that some of the information may be lost between consecutive frames.

Our general purpose was to create a system that would be able to show the differences in saccade parameters between different groups of disease severity of ND. Here we wanted to use the data and experience from previous studies, where we managed to successfully classify PD patients, healthy controls or people in different ages, based on results of their oculometric results

The system was also equipped with an estimator of facial emotions (FACS), because in previous works we proved the relationship between the chaosity in the eye movements and facial emotions such as Happiness [18, 19] and we wanted to create a tool allowing also for the continuation of the work related to this discovery, which will be described in the next article.

2 Methods

2.1 The conduction of oculometric tests

We have created a pipeline for real-time oculometric tests, which can be used in both, web browser-server solutions and in local implementations of desktop applications. Our idea was to enable all variants of taking measurements by using the computer of the examined person. We tested this solution for the capabilities with the RS. Our pipeline is based on a video stream from a web camera, and can be hosted in any application written in Python. The pipeline asynchronously processes images from the camera and for each video frame returning various information including position of the subject face, head position angles, position of the eyes center and current gaze direction. In addition, for each frame, information on the level of lighting is returned with a warning if it is not sufficient. Additionally, during calibration, the signal noise level is checked, which can be caused by various factors and which can also significantly affect the readings of eye movements and other parameters, this will be described later in this section. The pipeline also enables the detection and calculation of saccade parameters in 2 modes: real-time and post-processing. In the real-time mode, the current coordinates are buffered to a window whose size can be set depending on the desired calculation delay and the expected length of the saccade (depending on the distance between the fixation-point and the target). This mode can be useful for trial evaluation of patient saccades, calibration, or setting the distance between markers. In the post-process mode the entire record is subjected to the estimation of the components and the calculation of their parameters, which is usually more practical in the context of the analysis of the patient's results.

In current application, each test starts from head positioning when application shows a stream from a web camera with applied facial landmarks overlaid on the face and information for correction of head adjustments. In this regard, two parameters are calculated:

1. The distance between a head and the camera based on the distance between both eyes
2. The perpendicular head position determined by landmarks positions in the y-axis of the face bounding box.

The optimal distance from the camera was determined experimentally to be between 55 and 80 cm and. It allows the subject to sit close enough to be able to control the computer and not too far as an image of a face must be large enough. Then the next step was the calibration. We decided on a very basic 3-point calibration, because we found out during previous experiments that calibration is a problematic process especially for PD patients. On the other hand, we don't need to cast the entire screen in the physical space as the only purpose is to find fixation and the peripheral targets necessary for saccade stimulations. The fig.1 shows subject during calibration process (A) and the calibration schema (B).

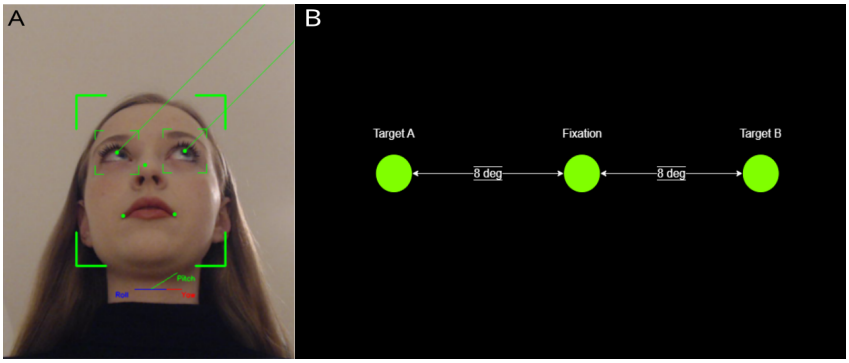


Fig. 1: Subject during calibration process (A) and the calibration process schema (B).

As mentioned before we used calibration to evaluate the noise of the signal and its frequency - thus its quality. The external factors such as insufficient lighting, reflections on the subject's glasses or poor concentration on stimulus can lead to low quality of the registration. We decided to use 2 parameters for noise level measurement: Signal to Noise Ratio (SNR) showing a relationship between means and statistical deviation (SD) in the signal and Periodogram of a signal as parameters, supporting judgment of signal noise level. During previous experiments, we established threshold values for acceptable noise level which are ≤ 3.5 for SNR and ≥ 0.02 average from the Periodogram with small tolerance for signals slightly different from both thresholds. Additionally we set frequency level to be >23 FPS (ideally 30 FPS , as stated in [20]) as in the course of our experimental work, we found that lower frequencies due to the increasing temporal sample error make impossible to calculate i.e. the saccade latency value within the acceptable error range, which is described in the next part of this text. The lightning condition is determined by calculating dominant color in a gray-scale image from the webcam with k-means clustering. We have experimentally determined that the average value obtained from the RGB of the dominant color indicating acceptable lightning conditions cannot be lower than 90. The complete schema of quality estimation process is presented in fig.2.

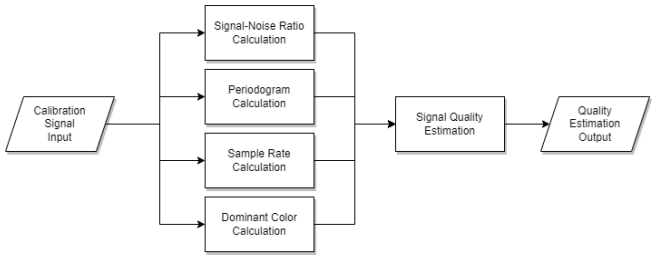


Fig. 2: Schema of a quality estimation process.

After successful calibration and checking the quality of the signal, the actual oculometric test is performed. Application displays the green ellipse in different locations dependently for fixation and peripheral targets. The fixation point shows up on the screen in a random interval between 1000 and 2000 milliseconds, then disappears in the same moment as the peripheral target appears on the left or right side of the fixation point in the same random time range. We set up the distance between fixation and peripheral markers for $\sim 8^{\circ}$ (fast, regular saccade). This sequence was conducted 10x during the whole test.

We used this "step model" of the measurement ($\Delta t = 0$) as the one in which differences in latency values between healthy controls and PD are the greatest [10], assuming that it would strengthen the differences between PD and healthy subjects. Prolonged latency in PD is related to both motor and cognitive impairment of the brain, but mostly related to bradykinesia [21] and is the main property differentiating saccades between healthy people and people suffering from the ND.

2.2 Pipeline estimations and calculations

We used modern approach of the inference provided by frameworks like Intel OpenVINO™ [22], an open-source toolkit for optimizing and boosting deep learning performance in computer vision [22]. In this approach Deep Learning models are deconstructed into hyper recursive functions (graphs) which allows to obtain results of neural network estimations with almost no delay for light models like Single Shot Detector (SSD). We also took advantage from Intel OpenVINO™ API and used asynchronous pipeline for parallel processing of the image frames. In this type of pipeline, images are received via an asynchronous queue, from where they are retrieved for processing by a parallel thread events, which allows for processing several frames at one time without blocking the main thread by waiting for synchronous results. We also used the OpenCV [23], the open-source library for image manipulations, for operations in image preprocessing (resizing, transpositions and reshaping into neural model input). For webcam video classifications and gaze predictions we used the pre-trained models [24] delivered with the Intel's OpenVINO™ Toolkit framework [22]. In order to estimate the gaze

position, in our pipeline we connected 4 pre-trained models, thus our estimation process consists of following 4 steps:

1. Face detection.
2. Facial Landmarks Estimation
3. Head Pose Estimation
4. Gaze Estimation

For face detection we used the lightweight "face-detection-adas-0001" model [25], based on the MobileNet architecture that includes depth-wise convolutions to reduce the amount of computation for the 3x3 block [25]. Its Average Precision (AP) was estimated to 0.94 for face bitmaps $\geq$100 px [25]. For facial landmarks estimation we used the "landmarks-regression-retail-0009" model [26]. This is a very lightweight model with a classic convolutional design, final regression is done by the global depthwise pooling head and fully connected layers with Mean Normed Error of 0.07 [26]. The model predicts only five facial landmarks: two eyes, nose, and two lip corners, which is enough for estimating eye's bounding box coordinates. For head pose estimation we used the "head-pose-estimation-adas-0001" model [27], an estimation network based on simple, CNN architecture with angle regression layers as convolutions, ReLU, batch normalization, fully connected with one output that represents value in Tait-Bryan angles (used to describe a general rotation in three-dimensional Euclidean space) with accuracy of: yaw 5.4 $\pm$ 4.4, pitch 5.5 $\pm$ 5.3 and roll 4.6 $\pm$ 5.6 [27]. For gaze estimation we used the "gaze-estimation-adas-0002" model [28], a custom VGG-like CNN for gaze direction estimation, which takes left and right eye images and head pose as an input [28]. It outputs a gaze directions prediction as 3-D vector in a Cartesian coordinate system in which the z-axis is directed from the mid-point between left and right eyes' centers to the camera center [28] with Mean Absolute Error of 6.95 $\pm$ 3.58 of angle in degrees [28].

The full gaze estimation process starts from the classification of the Face Estimation model and if a face is detected, the image processing continues in the pipeline. In current approach we took into account only the largest face bounding box, assuming that the face located closest to the screen is the examined subject. The image of the detected face is processed by the Facial Landmarks model to estimate positions of the eye sub-images. Simultaneously, the detected face image is processed by the Head Pose model to estimate Tait-Bryan angles of the head position. Results from both classifiers including the head "roll" angle are transferred to the model estimating gaze vectors. Finally, the gaze X,Y coordinates are calculated on the basis of the gaze vectors and the "roll" angle using trigonometric functions. Next we smoothed the signal using the low-pass butterworth 2nd order filter method implemented in the Scipy library with the cutoff=3.5 [29]. Smoothing was performed by omitting the data containing saccade peaks, pre-detected on the raw data. We just wanted to smooth out fragments of the signal from raw eye fluctuations without changing the shapes of the saccade amplitudes.

For saccade estimations we implemented an incremental rolling method that estimates the execution of the saccades in time series of gaze coordinates bas-

ing on stabilization, dispersion and changes in the amplitude. Algorithm first calculates eye fluctuation in the fixation state in the control window, which is set to around 300 ms experimentally by analyzing the registrations. The mean shift of the eye is then calculated from the difference in the x-axis coordinates of the control window. The start of the saccade search window is calculated from the moment when the peripheral target appears with the period of the possible minimal saccade latency assumed as 40 ms. From this starting point, the window end is set to around 500 ms, which covers the maximum saccade length for $\sim 8^{\circ}$. Then the window is searched for a movement of at least 50 ms (possible minimal duration for longer saccade) in the direction of the target direction. We assumed that if shift point between frames is greater than the mean shift in the control window (fixation state) and if the shift between the first and last frame of this sequence is $\geq 30\%$ of the distance between fixation and the peripheral target, the sequence meets our conditions and first frame can be set as the starting point of the saccade. We assumed the minimal saccade duration of 50 ms. After this interval algorithm looks for minimal inhibition of the eye movement and the frame in which shift relative to the previous will be $<30\%$ is considered as the saccade end point. If no starting or ending point was found during those sequences, we treated a trail as a failed or without saccade registration. For each detected saccade, we calculated its Latency, Duration, Amplitude, Average and Maximum Velocity and Gain (the ratio of saccade amplitude and fixation-target distance). The complete schema of the process is presented in fig.3.

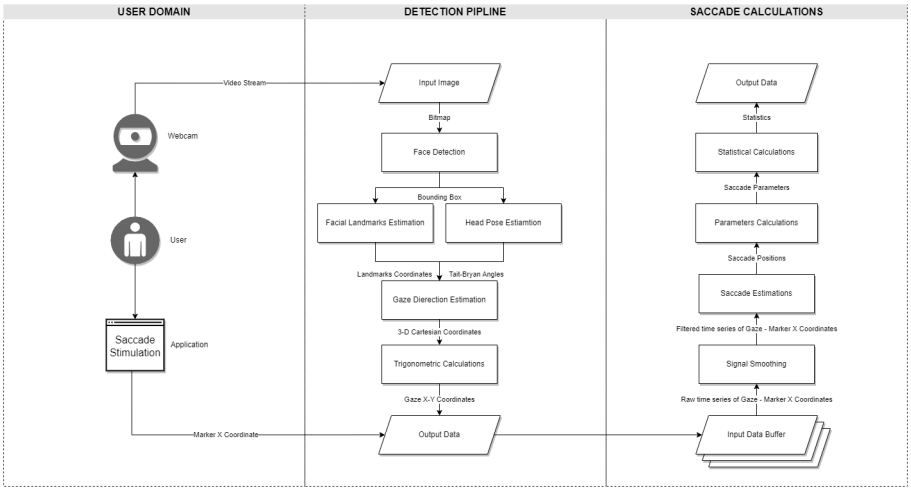


Fig. 3: Schema of a saccade estimation process.

3 Results

We compared the detection results of the implemented system with results obtained from infrared consumer-grade eye-tracker The Eye Tribe ETI1000 60Hz [30] also used in our previous experiments [31, 32]. For this purpose, we simultaneously registered saccadic tests with our pipeline using the webcam placed on top of the monitor and the ET placed on its bottom. The web camera used for this experiment was the Logitech C922 Pro Stream in Full HD and 30 Hz mode and the ET was running in 60Hz mode. We conducted the experiment in non-perfect lighting conditions, simulating non-laboratory environment. We obtained a recording from The Eye Tribe with actual frequency of 43 Hz, and from the webcam with frequency of 24 Hz. For the adequacy of the comparison, we first unified the time series between recordings, next, we resampled recordings from our pipeline to 43Hz with band-limited interpolation [33]. We also scaled both signals to a common range of values from -10 to 10 degrees, corresponding with an appropriate margin to distances between both peripheral targets. Then we used both signals as an input for our saccade detection and parameters calculations algorithms. Fig.4 presents the example of these recordings obtained simultaneously from The Eye Tribe and our pipeline. In this example the subject was a middle-aged man.

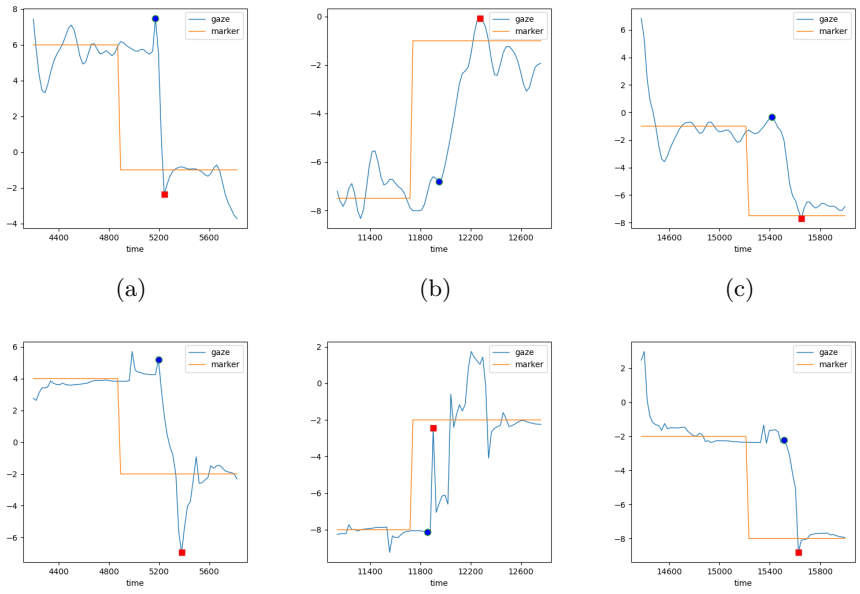


Fig. 4: The examples of horizontal 8°RS registrations from our pipeline with 30Hz webcam (a,b,c) and from The Eye Tribe ETI1000 60Hz (d,e,f).

From these graphs we can see that the signal from our system shows a smaller number of artifacts. We also detected a greater number of saccades (all 20 attempts were found). In case of saccade parameters, the mean latency obtained from the The Eye Tribe was 255.75 ± 60.74 and from our pipeline 245 ± 44.24 (∼10 ms, pvalue = 0.20). Statistically insignificant differences was also found for mean gain (The Eye Tribe = 1.04 ±0.14, our pipeline = 1.15 ±0.20, pvalue = 0.09) and for mean average velocity (The Eye Tribe = 70 deg/s ±36, our pipeline = 59 deg/s ±23, pvalue = 11.35).

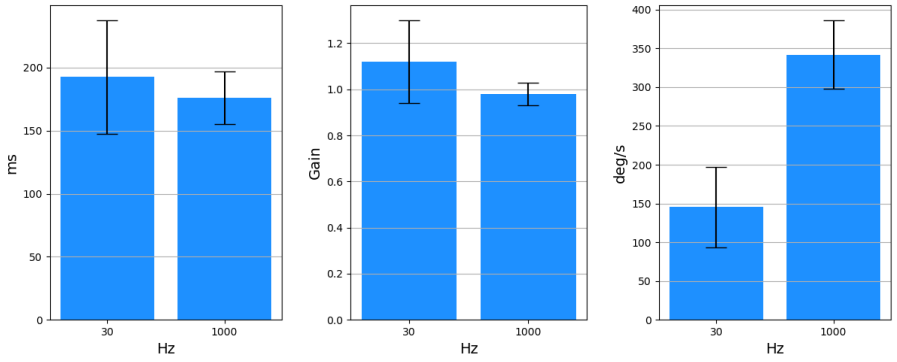
As we see, for these parameters, we have obtained confirmation of no significant differences between our pipeline and standard infrared eye-tracker with 2x higher frequency, unlike to the maximum velocity (approx. 71 ms difference, pvalue $\leq$ 0.05) which seems to be more sensitive to frequency changes in terms o the error rate, due to the temporal sampling error.

Additionally, we also compared our results with laboratory measurements for horizontal 8°RS (Table S2: "Descriptive and reproducibility results of the pro-saccadic task" - first measurements) done by using infrared eye tracker, The Eyelink 1000 Plus with frequency of 1000 Hz and published in [34]. The average age of the subjects in both examples was 40-41. In case of latency from our pipeline we obtained the mean of 192 ms ±45, while in the reference data mean was 176 ±21 and even these results are statistically significant (pvalue $\leq$ 0.05) the nominal difference between results was only 16 ms.

In case of mean gain it was 1.12 ±0.18, while in the reference data it was 0.98 ±0.05 [34], however in case of both types of velocity parameters we received differences between the reference data around 200 ms, far beyond the acceptable range, which confirms the impossibility of obtaining reliable results for these parameters in the case of lower frequencies. This inability comes directly from the bell-shaped characteristics of the velocity in ballistic type of movements, where the velocity increases and decreases linearly. In this characteristic the lower the sampling rate, the more information is lost about changes in velocity occurring between sampling points, so when eye increases velocity after the first sampling point (n) it results in change not being observed until another sample (n + 1) [35]. The bar charts on fig.5 presents statistical comparison for both acceptable results of latency and gain as well as for maximum velocity.

4 Discussions

Anderson et al. showed that one-point measures (which include RS) at low frequencies are burdened with a temporal sampling error derived from the fact that data for calculations are collected at a certain, finite frequency [35]. If frequency is lower than 1000Hz data i.e. of speed or direction are not registered between two consecutive measurements (n,n+1) and this error can affects also calculations of latency or duration. The size of the sampling error determines the difference between the time of the actual event (i.e. start of the saccade) and the time when the system actually registers the even occurrence. It makes actual oculomotor events uniformly distributed between sampling data points and the



(a) Latency (b) Gain (c) Maximum Velocity

Fig. 5: The comparison of horizontal 8°RS between 30Hz registrations from our pipeline and 1000Hz from the laboratory reference statistics [34]

expected mean of the sampling error i.e. at 50 Hz is 10 ms (1 ms error is the maximal one-point temporal sampling at 1000 Hz, increasing as the sampling frequency decreases) [35].

However, temporal sampling error has a different effect on different parameters, also depending on the type of the research. For example, for saccades smaller than 10°and the velocity parameters, sampling frequency should be >300 Hz to calculate it accurately [35–37]. But in the case of the latency (most important parameter for ND estimations) and in context of our research on ∼8°'fast regular' [38] saccades registered at 30Hz the difference in the result compared to the reference data of 1000Hz [34] was only 16 ms (20% of latency range 120-200 ms of healthy person). We expected this output because on average the sampling error will be in the middle of the sample interval, thus : $(1000/30)/2 = 16.6$ [35]. Interestingly, the outputs obtained from the ET showed similar difference between 10-20 ms and basing on results of those two experiments, we concluded the mean latency error to be inside this range. Such an error for analysis on difference between healthy subjects and PD patients has an marginal importance. Our previous studies on RS task with the 1000Hz eye-tracker, patients with PD showed latency ranges of (260.0, Inf) and (308.5, Inf) varying for specific disease stages [32, 39]. The healthy subjects for the same task show upper latency limit at around 190-200 ms [38]. In the context of such large differences, sampling error of 10-20 ms is unlikely to substantially affect the results of an overall data classifications.

We think that it is important to understand whether obtained sampling error magnitude brings problem in adequate scale to the research results. In case of this research, we assumed that our pipeline due to the limitations of consumer-grade equipment, will never compete in accuracy with laboratory equipment. However, it can provide reliable results for screening test that could allow specialist in specific cases to determine whether a more thorough examination is needed,

this time by using professional equipment such as a 1000 Hz eye-tracker in the laboratory conditions.

5 Conclusions

We propose to use a web cameras and oculometric pipelines like one presented in this text, where high frequency is not required and approximation of the results are enough to make an assumptions. Using latency as an example we showed that inaccuracy of calculations in low frequencies like 30Hz can be estimated at approx. 10-20 ms which gives minimal error rate in oculometric classifications of PD. If this rate is acceptable for your type of research, such machine learning solution based on neural networks and supporting algorithms can open possibilities to conduct tests without the need for expensive laboratory equipment.

Acknowledgements

Declaration of Competing Interest

Software frameworks and libraries: OpenVino, Tensorflow, OpenCV, SciPy, Numpy, Pandas, contributed to the development of the software methods presented in this article. The authors declare no conflict of interests.

References

1. Kilian Semmelmann and Sarah Weigelt. Online webcam-based eye tracking in cognitive science: A first look. *Behavior research methods*, 50, 06 2017.
2. Yu-Tzu Lin, Ruei-Yan Lin, Yu-Chih Lin, and Greg C. Lee. Real-time eye-gaze estimation using a low-resolution webcam. *Multimedia Tools and Applications*, 65:543–568, 2012.
3. Chunning Meng and Xuepeng Zhao. Webcam-based eye movement analysis using cnn. *IEEE Access*, 5:19581–19587, 2017.
4. Ahmad Aljaafreh, Murad Alaqtash, Naeem Al-Oudat, Jafar Abukhait, and Ma'En Saleh. A low-cost webcam-based eye tracker and saccade measurement system. 14, 04 2020.
5. Albert Śledzianowski, Artur Szymanski, Aldona Drabik, Stanisław Szlufik, Dariusz M. Koziorowski, and Andrzej W. Przybyszewski. Combining results of different oculometric tests improved prediction of parkinson's disease development. In Ngoc Thanh Nguyen, Kietikul Jearanaitanakij, Ali Selamat, Bogdan Trawiński, and Suphamit Chittayasothorn, editors, *Intelligent Information and Database Systems*, pages 517–526, Cham, 2020. Springer International Publishing.
6. Albert Sledzianowski, Artur Szymanski, Aldona Drabik, Stanisław Szlufik, Dariusz M. Koziorowski, and Andrzej W. Przybyszewski. Measurements of antisaccades parameters can improve the prediction of parkinson's disease progression. In Ngoc Thanh Nguyen, Ford Lumban Gaol, Tzung-Pei Hong, and Bogdan Trawiński, editors, *Intelligent Information and Database Systems*, pages 602–614, Cham, 2019. Springer International Publishing.

7. Albert Śledzianowski, Artur Szymański, Stanisław Szlufik, and Dariusz Koziorowski. Rough set data mining algorithms and pursuit eye movement measurements help to predict symptom development in parkinson's disease. In Ngoc Thanh Nguyen, Duong Hung Hoang, Tzung-Pei Hong, Hoang Pham, and Bogdan Trawiński, editors, *Intelligent Information and Database Systems*, pages 428–435, Cham, 2018. Springer International Publishing.
8. Andrzej Przybyszewski, Mark Kon, Stanislaw Szlufik, Artur Szymański, Piotr Habela, and Dariusz Koziorowski. Multimodal learning and intelligent prediction of symptom development in individual parkinson's patients. *Sensors*, 16:1498, 09 2016.
9. Andrzej W. Przybyszewski, Stanislaw Szlufik, Justyna Dutkiewicz, Piotr Habela, and Dariusz M. Koziorowski. Machine learning on the video basis of slow pursuit eye movements can predict symptom development in parkinson's patients. In Ngoc Thanh Nguyen, Bogdan Trawiński, and Raymond Kosala, editors, *Intelligent Information and Database Systems*, pages 268–276, Cham, 2015. Springer International Publishing.
10. Jonathan M. Chambers and Tony J. Prescott. Response times for visually guided saccades in persons with parkinson's disease: A meta-analytic review. *Neuropsychologia*, 48(4):887–899, 2010.
11. Travis Turner, Jenna Renfroe, Amy Delambo, and Vanessa Hinson. Validation of a behavioral approach for measuring saccades in parkinson's disease. *Journal of motor behavior*, 49:1–11, 01 2017.
12. Samuel Stuart, Rachael A. Lawson, Alison J. Yarnall, Jeremy Nell, Lisa Alcock, Gordon W. Duncan, Tien K. Khoo, Roger A. Barker, Lynn Rochester, David J. Burn, and on behalf of the ICICLE-PD study group. Pro-saccades predict cognitive decline in parkinson's disease: Icicle-pd. *Movement Disorders*, 34(11):1690–1698, 2019.
13. Robert Perneczky, Boyd C.P. Ghosh, Laura Hughes, Roger H.S. Carpenter, Roger A. Barker, and James B. Rowe. Saccadic latency in parkinson's disease correlates with executive function and brain atrophy, but not motor severity. *Neurobiology of Disease*, 43(1):79–85, 2011. Autophagy and protein degradation in neurological diseases.
14. Chrystalina Antoniades, Zheyu Xu, Roger Carpenter, and Roger Barker. The relationship between abnormalities of saccadic and manual response times in parkinson's disease. *Journal of Parkinson's disease*, 3, 10 2013.
15. Qing Yang, Tao Wang, Ning Su, Shifu Xiao, and Zoi Kapoula. Specific saccade deficits in patients with alzheimer's disease at mid to moderate stage and in patients with amnestic cognitive impairment. *Age (Dordrecht, Netherlands)*, 35, 05 2012.
16. Marta Pereira, Martina Villa, Do Hyong Koh, Marina von Zuben de Arruda Camargo, Ariella Belan, Márcia Radanovic, Marc Pomplun, and Orestes Forlenza. Saccadic eye movements associated with executive function decline in mild cognitive impairment and alzheimer's disease: Biomarkers (non-neuroimaging) / novel biomarkers. *Alzheimer's And Dementia*, 16, 12 2020.
17. Adam Boxer, Siobhan Garbutt, William Seeley, Aria Jafari, Hilary Heuer, Jacob Mirsky, Joanna Hellmuth, John Trojanowski, Erik Huang, Steven DeArmond, John Neuhaus, and Bruce Miller. Saccade abnormalities in autopsy-confirmed frontotemporal lobar degeneration and alzheimer disease. *Archives of neurology*, 69:509–17, 04 2012.

18. Albert Sledzianowski, Krzysztof Urbanowicz, Wojciech Glac, Renata Slota, Maria Wojtowicz, Monika Nowak, and Andrzej Przybyszewski. Face emotional responses correlate with chaotic dynamics of eye movements. *Procedia Computer Science*, 192:2881–2892, 2021. Knowledge-Based and Intelligent Information And Engineering Systems: Proceedings of the 25th International Conference KES2021.
19. Albert Śledzianowski, Jerzy P. Nowacki, Andrzej W. Przybyszewski, and Krzysztof Urbanowicz. Detecting true and declarative facial emotions by changes in nonlinear dynamics of eye movements. In Ngoc Thanh Nguyen, Tien Khoa Tran, Ualsher Tukayev, Tzung-Pei Hong, Bogdan Trawiński, and Edward Szczerbicki, editors, *Intelligent Information and Database Systems*, pages 106–116, Cham, 2022. Springer International Publishing.
20. Jacek Naruniec, Michał Wieczorek, Stanisław Szlufik, Dariusz Koziorowski, Michał Tomaszewski, Marek Kowalski, and Andrzej Przybyszewski. Webcam-based system for video-oculography. *IET Computer Vision*, 11(2):173–180, 2017.
21. Michael R. MacAskill, Charlotte F. Graham, Toni L. Pitcher, Daniel J. Myall, Leslie Livingston, Saskia van Stockum, John C. Dalrymple-Alford, and Tim J. Anderson. The influence of motor and cognitive impairment upon visually-guided saccades in parkinson's disease. *Neuropsychologia*, 50(14):3338–3347, 2012.
22. Intel®. Intel® distribution of openvino™ toolkit, https://docs.openvino.ai/, 2022.
23. G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.
24. Intel®. Openvino™ toolkit - open model zoo repository, intel's pre-trained models, https://github.com/openvinotoolkit/open_model_zoo, 2022.
25. Intel®. Openvino™ toolkit - open model zoo repository, intel's pre-trained models https://docs.openvino.ai/latest/omz_models_model_face_detection_adas_0001.html, 2022.
26. Intel®. Openvino™ toolkit - open model zoo repository, intel's pre-trained models https://docs.openvino.ai/latest/omz_models_model_landmarks_regression_retail_0009.html, 2022.
27. Intel®. Openvino™ toolkit - open model zoo repository, intel's pre-trained models https://docs.openvino.ai/2019_r1/_head_pose_estimation_adas_0001_description_head_pose_estimation_adas_0001.html, 2022.
28. Intel®. Openvino™ toolkit - open model zoo repository, intel's pre-trained models https://docs.openvino.ai/2019_r1/_gaze_estimation_adas_0002_description_gaze_estimation_adas_0002.html, 2022.
29. Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020.
30. The EyeTribe. The eyetribe documentation and api reference, https://github.com/eyetribe/documentation, 2022.
31. Albert Sledzianowski. Predictions of age and mood based on changes in saccades parameters. In Aleksandra Gruca, Tadeusz Czachórski, Sebastian Deorowicz, Katarzyna Harezlak, and Agnieszka Piotrowska, editors, *Man-Machine Interactions 6*, pages 196–205, Cham, 2020. Springer International Publishing.

32. Artur Szymanski, Stanisław Szlufik, Dariusz M. Koziorowski, and Andrzej W. Przybyszewski. Building classifiers for parkinson's disease using new eye tribe tracking method. In *ACIIDS*, 2017.
33. Julius O. Smith. *Digital Audio Resampling Home Page*. https://ccrma.stanford.edu/ jos/resample/, January 28, 2002.
34. Jenny Nij Bijvank, A. Petzold, Lisanne Balk, H. Tan, B. Uitdehaag, M. Theodorou, and L. Rijn. A standardized protocol for quantification of saccadic eye movements: Demons. *PLOS ONE*, 13:e0200695, 07 2018.
35. Richard Andersson, Marcus Nyström, and Kenneth Holmqvist. Sampling frequency and eye-tracking measures: How speed affects durations, latencies, and more. *Journal of Eye Movement Research*, 3:1–12, 09 2010.
36. J. T. Enright. Estimating peak velocity of rapid eye movements from video recordings. *Behavior Research Methods, Instruments, and Computers*, 2:1532–5970, 06 1998.
37. Pyykko I. Juhola M., Jantti V. Effect of sampling frequencies on computation of the maximum velocity of saccadic eye movements. *Biological Cybernetics*, 53:1432–0770, 12 1985.
38. Ramsperger E. Fischer B. Human express saccades: extremely short reaction times of goal directed eye movements. *Exp Brain Res.*, 57(1):191–5, 1984.
39. Andrzej Przybyszewski, Mark Kon, Stanislaw Szlufik, Artur Szymański, Piotr Habela, and Dariusz Koziorowski. Multimodal learning and intelligent prediction of symptom development in individual parkinson's patients. *Sensors*, 16:1498, 09 2016.