2023 IEEE 24th International Conference on Information Reuse and Integration for Data Science (IRI) | 979-8-3503-3458-6/23/$31.00 ©2023 IEEE | DOI: 10.1109/IRI58017.2023.00008

Microsaccade Segmentation using Directional Variance Analysis and Artificial Neural Networks

Shan Suthaharan
Department of Computer Science
University of North Carolina at Greensboro
Greensboro, NC, USA
s_suthah@uncg.edu

Daniel M.W. Lee
Department of Bioengineering
University of Pittsburgh
Pittsburgh, PA, USA
dal208@pitt.edu

Min Zhang and Ethan A. Rossi
Department of Ophthalmology
University of Pittsburgh
Pittsburgh, PA, USA
{miz62, rossiea}@pitt.edu

***Abstract*—Fixational eye movements (FEMs) are an essential component of vision and there is considerable research interest in using them as biomarkers of brain injury and neurodegeneration. Study of FEMs often involves segmenting them into their individual components, primarily microsaccades and drifts. In practice, velocity (or acceleration) thresholds are commonly adapted–while they are generally imperfect–requiring tuning of thresholds and manual correction and verification by human graders. Manual segmentation and correction is a tedious and time-consuming process for human graders. Fortunately, it can be observed from Tracking scanning laser ophthalmoscopy (TSLO) video recordings that the directional variances of FEMs can be extracted to mathematically characterize microsaccades for segmentation and distinguish from drift. Therefore, we perform a directional variance analysis, extract relevant features, and automate the model using artificial neural networks (ANN). We propose and compare two directional variance approaches along with an ANN model for the segmentation of microsaccades. The first approach utilizes a single-point based feature variance, whereas the second approach utilizes a sliding-window based feature variance with the information from several time points. We calculate several statistical metrics to characterize the features of the microsaccades such as the number of microsaccades, microsaccade peak velocity and acceleration, and microsaccade duration. We have also calculated the accuracy, precision, sensitivity, and specificity scores for each approach to compare their performance. The single-point models labeled the FEM data with an accuracy of 70% whereas the sliding-window approach had an accuracy of 85%. When comparing the percent errors of the approaches to the ground truth, the sliding-window approach performs significantly better than the single-point approach as it captures more relevant directional variance features of FEMs.**

***Index Terms*—Fixational eye movement, microsaccade, artificial neural networks, directional variance, feature space**

I. Introduction

Precise quantification of fixational eye movements offers promise towards improved understanding and detection of neurological conditions and brain injury [1]. Fixational eye movements (FEMs) are the miniature, involuntary eye movements that keep the eye in constant motion, even when an individual attempts to maintain their gaze on a motionless target [2]. There are two main classes of FEMs: microsaccades and ocular drifts. In brief, microsaccades are fast, ballistic movements whereas drifts are slow aperiodic movements that occur between microsaccades [3]. The majority of fixation time is spent in the drift state. FEMs are essential not only for visual function but also neural processing. Enhancing spatial vision and synchronizing sensory and motor neural activity are just two purposes that FEMs serve [3], [4].

Current clinical approaches to measure eye movements have utilized video-based eye-trackers that in general lack the resolution to precisely measure the smallest movements of the eye [5], [6]. With TSLO, clinicians and researchers may obtain precise FEM data with higher spatial and temporal resolution compared to other eye trackers [7]. The TSLO records movement of the retina and the FEM data can be extracted using custom software [4]. With the extracted FEM data, using a simple threshold on a parameter such as velocity or acceleration may suffice for a baseline classification for microsaccades but due to their complex nature, being able to accurately and consistently pinpoint the starts and ends of them with this method remains difficult. The inaccuracies of temporal measurements make the calculation of metric (e.g., acceleration, duration and amplitude) more difficult.

Precise determination of the entire timecourse of each microsaccade cannot be determined using this method so require alternative methods, such as the manual marking of the starts and ends of the microsaccades. Also, due to the lack of any sophisticated method of accurately marking the starts and ends of microsaccades, the only alternative is to proceed with the cumbersome process of manual marking. Previous work has been carried out to minimize manual classification of micosaccades. Engbert and Kliegl pioneered this area of research on undergraduate students fixating on a target [8]. The detection threshold adapted was based on the standard deviation of the velocity. They also used binocular information to reduce noise during the detection process. A unique feature used in this method is a variable ”lambda” factor that is influenced by the noise present in the dataset. This allows for robustness across subjects but also introduces another parameter that needs to be adjusted between trials.

Otero-Millan, Castro, Macknik and Martinez-Conde developed a method of eye movement detection using an unsupervised learning method clustering technique [9]. The clustering technique groups observations with similar features. For this method, they considered features such as peak velocity and acceleration when detecting microsaccades. They used principal component analysis on the characteristics that were highly

979-8-3503-3458-6/23/$31.00 ©2023 IEEE
DOI 10.1109/IRI58017.2023.00008

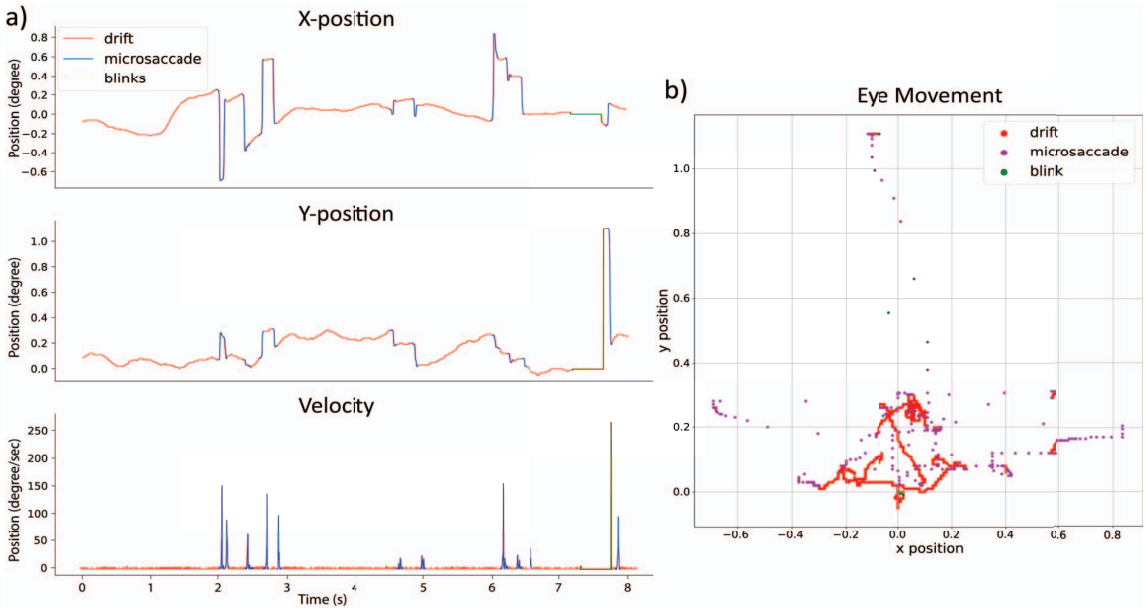

Fig. 1. (a) Graph that displays the x-position, y-position, and velocity of a participant's eye motion trace with respect to time. (b) Lateral movement of eye that depicts a vector space–determined by x-position, y-position, and velocity–in which a vector belongs to one of the groups (microsaccade, drift, or blink).

correlated and then clustered to decrease dimensions and computation costs. Zemblys et al. have studied and developed a fully automated classification of raw gaze samplings as belonging to fixations, saccades, and post-saccadic oscillations by training a random forest (RF) classifier [10]. They trained their classifier using ground truth labels and several post-processing heuristics. They then assessed the performance of their classifier and found that it was able to detect all saccades and subsequent post-saccadic oscillations. However, the microscopic measurements recorded by a TSLO, such as microsaccades, will be different from large gaze-orienting saccades in that they are involuntary, rapid, and much smaller in magnitude. These methods were consistent in using peak velocity and acceleration as key features used to classify the microsaccades. However, it is expected that the artificial neural network (ANN) architecture–due to its sophisticated feature learning module–could efficiently learn the features of microsaccades dynamically from the data by performing a simple threshold on these features. Motivated by the ability of classification of a one-dimensional ANN architecture [11]–[15], we present an experimentally validated neural network model that is capable of automatically segmenting microsaccades from drifts. We also proposed two different approaches that differ in the feature space of the dataset used.

The ANN model is a computational tool that is capable of predicting response variables from a dataset to make decisions. Its architecture is inspired by the neuronal connections within an animal brain [16]. Similar to the brain, there are different layers in an ANN and each layer has varying numbers of nodes, which are analogous to neurons in the brain. When backpropagating through each layer, an output weight is computed through an activation function that can be tuned by the user [16]. There are a myriad of hyperparameters in an ANN that can be fine-tuned by the user to improve the performance. The number of nodes in the hidden layers and the number of hidden layers are the hyperparameters that were tuned in this study. The tuning of these parameters were made through through a sequence of simulations by treating ANN's architecture a black box. In this study, we present an ANN model that was trained using feature vectors that are generated by using two different feature space approaches.

II. Methods

The first approach used a single-point based feature space. In this approach, the model learned to label each FEM data point based on the information of the features from a single data point. The second approach used a sliding-window based feature space. In this case, the model learned to label each FEM data point given the information of the features from not only one data point but also several data points before and after. The preprocessing protocol for the training data differed across the two approaches. We worked with a time-series dataset of FEM recordings from healthy participants imaged in a previous study [17].

A. Type of Data Used

The FEM datasets used came from 44 different participants (mean age 17.0 $\pm$ 3.2 years; 24 females (55%)). Multiple recordings (varying between 1-10 recordings per participant) were acquired from each participant; a total of 340 TSLO recordings was included in this study. The motion of the eye was extracted from these videos using a custom software

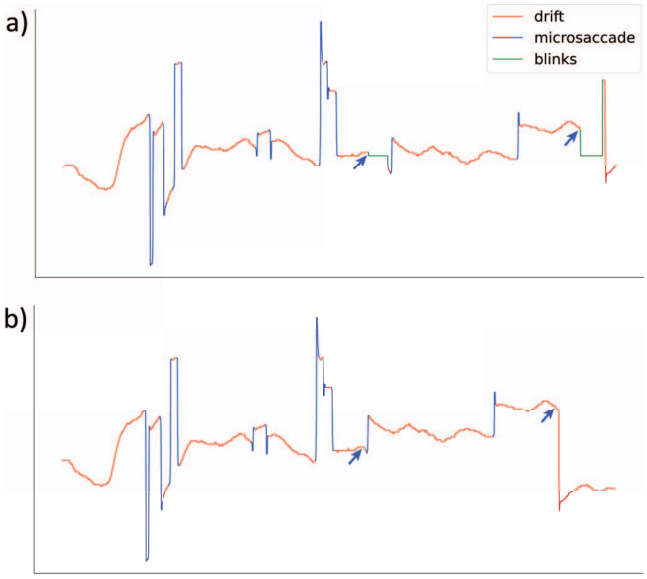

Fig. 2. (a) Eye motion trace of eye movement with blinks, (b) Eye motion trace of eye movement without blinks.

that was developed at Rossi Lab [18]. The eye motion information that as extracted from these 30 frames per second recordings using 16 strips per frame resulted in a nominal temporal sampling frequency of 480 Hz. Microsaccades, were segmented using a tedious manually intensive approach where a velocity threshold was used to identify microsaccades and their start and end points were marked manually by trained graders. The dataset provided an x and y positions, and a label (microsaccade or drift) at each time point, see Fig. 1.

B. Preprocessing of Data

The data collected from all of the participants were concatenated to create one large dataset. We removed the blinks from the compiled dataset by locating them using their corresponding labels. The blinks were removed by adapting the method proposed in [18]. However, removing the blinks and simply concatenating the data points before and after the blinks creates an issue of discontinuous data. The positions of the eye before and after blinks are not necessarily the same, resulting in the data showing a sudden change in eye position. Due to microsaccades being classified as fast, ballistic movements, the neural network may misclassify this discontinuity as a microsaccade. We incorporated a stitching function to avoid this type of misclassification error. The function calculates the difference in position between the data points before and after the blink and sums all data points subsequent the blink, resulting in the dataset appearing continuous. A representative eye trace before and after applying the stitching function is shown in Fig. 2. Once all blink labels were removed from the dataset, the velocity of the eye was calculated using the derivative of the positional information. The velocity of the first data point was assumed to be 0 for all datasets.

C. Computational Modeling

We denoted the preprocessed time series data of FEM recordings by $U = \{u_1, u_2, \ldots, u_n\}$, where $u \in U$ is a three-dimensional data vector $u = (x, y, v)$ and u is labeled as either a drift (1) or a microsaccade (0). The variables x, y, and v, as stated before, represent the x-position, y-position, and the velocity of the eye. We then constructed two types of feature vectors to characterize the relationship between their feature spaces and the response sets (labels 0 or 1). This was implemented with two computational models for developing machine learning techniques. The core concept adapted in these computational model is the *directional variance*.

1) Directional Variance: The concept of directional variance has been used in many applications, including the design of an image quality metric [19], fingerprint classification [20], and [21]. To capture the directional variances of the feature vector (x, y, v) with respect to microsaccade, drift, and blink, we constructed feature vectors (or space) using single-point directional variance analysis and sliding-window directional variance analysis. This is also called feature learning of FEMs.

2) single-point Feature Learning: We modeled the feature space for the single-point analysis using the three-dimensional vector $u = (x, y, v)$ such that a parametric learning model $f_{\alpha,\beta}$ that approximates the following mathematical relationship becomes our machine learning model:

$$l = f_{\alpha,\beta}(u), \tag{1}$$

where the parameters α and β are sets of hyperparameters and learning parameters, and $l \in \{0, 1\}$.

3) Sliding-Window Feature Learning: We modeled the feature space for the sliding-window analysis using the window of consecutive data points $(d_1, d_2, \ldots, d_k)$ in the time series data of FEM recordings by U as its feature vector. In other words, the step size of the sliding window is a single data point to identify and record microsaccade labels that are generally sparse. Although this process generates a large feature space with many redundant observations, the modern computing resources and the automated ML classifiers allow the proposed approach to handle such a big data environment. Each data point is a three-dimensional feature vector (x, y, v); hence, this feature vector is a $3k$ dimensional vector. We represent this feature vector by $w_k = (w_1, w_2, \ldots, w_{3k})$. Note that we empirically determined that $k = 12$ was sufficient to define the drift or microsaccade classes; hence, the dimension of the feature vector used for the sliding-window analysis in our experiments is 36. Therefore, our learning model is:

$$l = f_{\alpha,\beta}(w_k), \tag{2}$$

where the parameters α and β are sets of hyperparameters and learning parameters, and $l \in \{0, 1\}$. Additionally k is also considered as a hyperparameter. The dataset for the sliding-window analysis consisted of 37 elements in each data point. The size of the window was set to 12. The x position, y position, and velocity for the 12 data points were in each window. If the window contained more drifts than

microsaccades, that window was labeled as a microsaccade and vice versa. All ties were labeled as microsaccades.

D. Balancing the Feature Spaces

Imbalanced datasets can be problematic for training of neural networks. The dataset was imbalanced with 92.1% drifts and 7.9% microsaccades. With this imbalanced data characteristics, the ANN model could label all datapoints as microsaccades and achieve 92.1% accuracy, without learning the features of microsaccades. To solve this issue, we separated the microsaccades and drifts into two different datasets. We then created new, balanced datasets by combining all the microsaccades with the equivalent number of data points of drifts. The drift data points used on a previous balanced dataset were not reused when creating another balanced dataset. This process was repeated until the number of drifts in the original drift dataset was less than the total number of microsaccades in the microsaccade dataset. This resulted in 11 datasets with an even distribution of microsaccades and drifts. The 11 datasets were then randomized and split into the training and testing datasets (80:20). Using these datasets, 11 models were trained and their performances were compared to test for consistency.

E. ANN Architecture

The proposed ANN contained 7 hidden layers and 2 dropout layers: one between the third and fourth layer and the second between the seventh and output layer. The proposed number of hidden layers was determined empirically to avoid overfitting, due to the large number of connections, but also to maximize efficiency by keeping training time to a minimum. The activation functions for the hidden layers and the output layer were ReLU and sigmoid, respectively. The dropout rate for the dropout layers was set to 0.5 to avoid the propagation of bias between any two gradient descent operations that could negatively impact the distinguishable features between microsaccade and drift. The number of nodes was set to 8192, 4096, 2048, 1024, 512, 256, and 128 in that order from the first to seventh hidden layer. The number of nodes and dropout layers, and the dropout rates were determined through trial and error to maximize the model performance, avoid overfitting to the training data, and minimize training time.

The same neural network architecture was utilized for both the single-point and sliding-window approaches. This was carried out to ensure the changes in performance is due to the feature learning approach and not the architecture. After the models were trained, the unlabeled testing data were tested to analyze the performance of the models. We calculated the accuracy, precision, sensitivity, and specificity [11] to measure the performance of the ANN models. Other metrics, the number of microsaccades and the average peak velocity, acceleration, duration and amplitude of the microsaccades were also calculated. The metrics were compared with the ground truth to compare the performance of the two approaches.

III. Results

The performance of the trained ANN models were evaluated using the test data. Fig. 3 compares the classification of microsaccades and drifts for the ground truth, sliding-window approach, and single-point approach. The sliding-window approach performed better than the single-point approach. As shown in Fig. 3(b) and Fig. 3(c), the sliding-window approach was able to detect and segment all microsaccades in the dataset and additional microsaccades were labeled as well (later discussed). However, the single-point approach performed poorly, not being able to detect even half the microsaccades in this particular dataset. Fig. 3 mainly compares the labels generated by the two approaches. Statistical metrics were calculated to further analyze the performance of both approaches.

TABLE I
CONFUSION MATRIX FOR BOTH FEATURE SPACE APPROACHES.

Performance Metrics	**Performance Scores**	
	single-point Analysis	Sliding-Window Analysis
Accuracy	69.8	84.5
Precision	79.6	90.3
Sensitivity	53.3	77.9
Specificity	86.2	91.2

The confusion matrix was calculated for the 11 different models of the two approaches and the averages were taken to produce one confusion matrix for both approaches. Table 1 displays the statistical metrics calculated for both approaches averaged across the 11 models that were trained for each approach. For this study, the microsaccades were considered as positive in the confusion matrix and the drifts were considered as negative. The sliding-window approach had an accuracy score of 84.5% whereas the single-point approach had an accuracy score of 69.8%. The precision, sensitivity and specificity scores for the sliding-window approach were all greater than that of the single-point approach. The difference shown in Table 2 demonstrates that the sliding-window approach was able to outperform the single-point approach as it was closer to the ground truth when comparing the percentage errors of the two feature space approaches. The only metric that was not too different was the amplitude but the rest showed superiority of the sliding-window approach. These percentage errors provide support that the sliding-window approach substantially outperformed the single-point approach.

The precision scores of 79.6 and 90.3 indicate that our trained 1D ANN models with the proposed feature learning approaches are efficient and can segment (or detect) microsaccades with very low false positive and high true positive measures in both feature-extraction schemes. These quality measures also indicate that the sliding-window feature learning is highly efficient. At the same time, the sensitivity score of 53.3 explains that the false negative and true positive are almost the same, which indicate the model has some performance issue in classifying the drift class under the single-point analysis. However, the larger sensitivity score of 77.9 of the sliding-window analysis indicate that the model can classify the drift class in the sliding-window model.

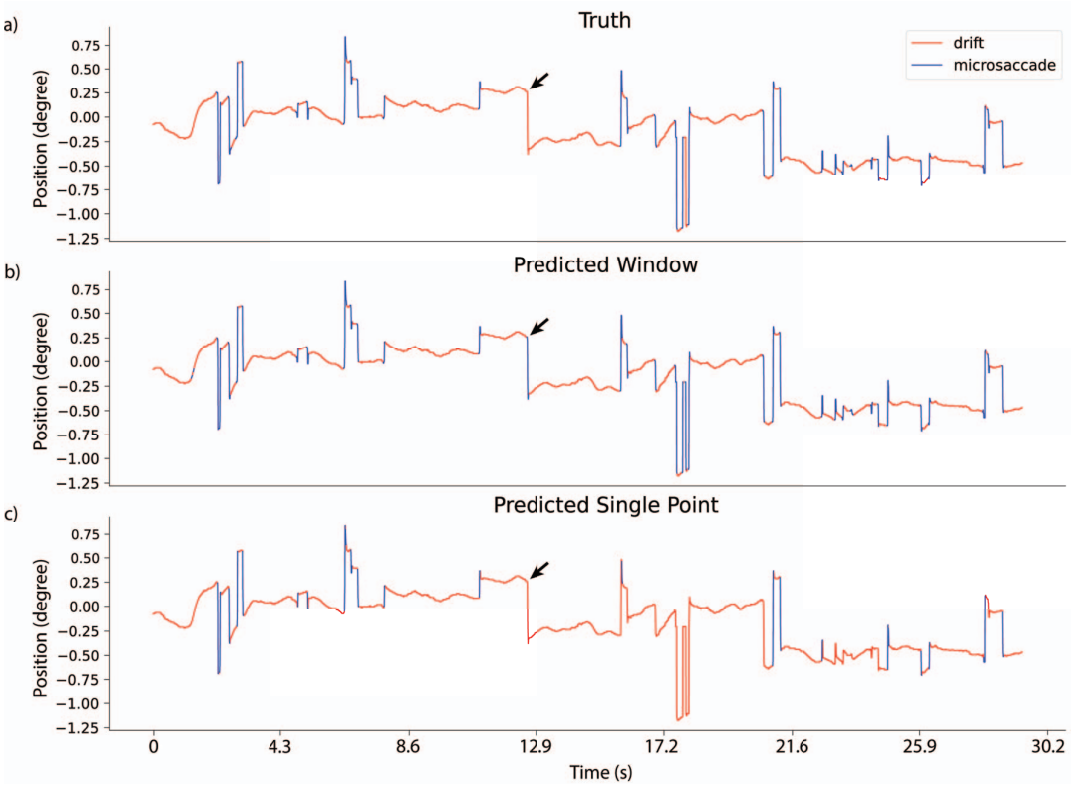

Fig. 3. **Comparison between ground truth and ANN using either the single-point or sliding-window feature space for training**: A dataset was passed through the single-point and sliding-window approaches to be labeled. The figure above shows the classification in the x-position parameter space. Top: ground truth, Middle: sliding-window approach, Bottom: single-point approach. The arrows show a discrepancy in classification between the sliding-window approach compared to both the ground truth and single-point approach.

TABLE II
THE AVERAGE PERCENT ERROR ACROSS THE 22 TEST FEM DATASETS WERE CALCULATED FOR THE FIVE STATISTICAL METRICS. THE ERROR WAS COMPARING THE SINGLE-POINT APPROACH AND THE SLIDING-WINDOW APPROACH TO THE GROUND TRUTH.

Statistical Metrics	**Percent error (%)**									
	Number of Microsaccades		**Average Peak Velocity**		**Average Peak Acceleration**		**Average Duration**		**Average Amplitude**	
Approach	Single Point	Sliding Window	Single Point	Sliding Window	Single Point	Sliding Window	Single Point	Sliding Window	Single Point	Sliding Window
Average	67.2	21.6	25.2	13.8	34.5	14.9	50.9	27.1	15.3	13.1

IV. DISCUSSION

The sliding-window analysis approach outperformed the single-point analysis approach when considering all the statistical metrics. More specifically, the precision and specificity values for the sliding-window approach were higher than that of the single-point approach, indicating low false positives with high true positives and true negatives. This also indicates that the model that was trained with the sliding-window based feature space was superior to the single-point based feature space. The percent errors also provide more support that the sliding-window based feature space was superior to the single-point based feature space since the sliding-window approach had significantly lower errors in all metrics compared. An interesting finding can be seen in Fig.3, comparing the ground truth and the sliding-window approach. The sliding-window approach was able to capture all of the microsaccades that were labeled by the ground truth but there were other data points labeled as a microsaccade where the ground truth label was a drift (arrows in Fig. 3). Since manual labeling is subjective and prone to human error, the grader may have missed this microsaccade during grading. Hence, we revisited the data and carefully inspected the raw TSLO video to observe the grading errors. The video clearly showed that there was a microsaccade at this position, demonstrating that the trained ANN detected a microsaccade that was missed by the grader. This is just one instance of this finding. There are other instances where the trained ANN picked up on a few missed microsaccades. We propose two reasons for these unmarked microsaccades: the trained ANN model mislabeled a microsaccade that should have been actually labeled as

drift or it detected a microsaccade that a grader might have missed or did not consider to be a microsaccade. This leads to the possibility that a trained ANN model not only grade the FEM dataset quicker than graders but also be more accurate. It important to study this problem further to confirm this findings. The architecture of the ANN is simple and works well for the purposes of this study, but there are ways to improve the neural network to further increase accuracy and performance. The feature space used in this study consists of the two-dimensional positions, and the velocity. Increasing the feature space is expected to improve the performance.

There is a study that utilized a one-dimensional convolutional neural network (CNN) to classify human activity (https://machinelearningmastery.com/cnn-models-for-human-activity-recognition-time-series-classification/). This is an alternative that can provide improved results as a CNN may be able to learn the complex, unique behaviors of microsaccades better than an ANN model. Since we aim to have a method for microsaccade segmentation that is robust and can be used as a first step towards the extraction of diagnostic features from eye movement recordings, it is essential that it be as close to flawless in its performance as possible. Our future research will include the development of a fully automated microsaccade segmentation approach. Such a tool would facilitate the use of FEMs in clinical settings, where rapid segmentation and quantification of microsaccade parameters might be useful for evaluating patients. The ANN model also needs to be streamlined into clinical devices so that clinicians can have access to these metrics during clinical visits from patients. It will allow for immediate interpretation of the patient's results. We recently demonstrated that microsaccades can differentiate concussed patients from controls in a certain type of fixation task [17]. However, analyzing the properties of microsaccades is just the first step towards using FEM recordings as a diagnostic tool. Segmented eye traces would undergo subsequent evaluation to determine if a patient is a healthy patient or an injured (or a diseased) patient.

V. Conclusion

Automated microsaccade segmentation can be successfully achieved using a one-dimensional ANN with a sliding-window based feature space that is enhanced by the directional variance measure. This approach may be useful for the automatic segmentation of microsaccades in clinical settings. Efficient and robust automatic extraction of FEMs based biomarkers offers promise as a powerful tool for improved detection and monitoring of neurological conditions and brain injury.

Acknowledgment

This research work was supported by departmental startup funds from the University of Pittsburgh to Ethan A. Rossi. This research project was also supported by NIH CORE Grant P30 EY08098 to the University of Pittsburgh Department of Ophthalmology and from an unrestricted grant to the University of Pittsburgh Department of Ophthalmology from Research to Prevent Blindness, New York, NY, USA.

References

[1] R. G. Alexander, S. L. Macknik, and S. Martinez-Conde, "Microsaccade characteristics in neurological and ophthalmic disease," *Frontiers in Neurology*, vol. 9, 2018.

[2] S. Martinez-Conde, J. Otero-Millan, and S. L. Macknik, "The impact of microsaccades on Vision: Towards a unified theory of saccadic function," *Nature Reviews Neuroscience*, vol. 14, no. 2, pp. 83–96, 2013.

[3] M. Rucci and M. Poletti, "Control and functions of Fixational Eye Movements," *Annual Review of Vision Science*, vol. 1, no. 1, pp. 499–518, 2015.

[4] M. Rucci, R. Iovin, M. Poletti and F. Santini (2007). "Miniature Eye movements enhance fine spatial detail," *Nature*, 447(7146), 852–855. https://doi.org/10.1038/nature05866 *Biomedical Optics Express*, vol. 12, no. 4, pp. 2353, 2021.

[5] M. Nyström, D. C. Niehorster, R. Andersson, R. S. Hessels and I. T. Hooge, "The amplitude of small eye movements can be accurately estimated with video-based eye trackers," Behavior Research Methods, pp. 1-13, 2022.

[6] K. Holmqvist and P. Blignaut, "Small eye movements cannot be reliably measured by video-based P-CR eye-trackers," Behavior research methods, 52, pp. 2098-2121, 2020.

[7] C. K. Sheehy, Q. Yang, D. W. Arathorn, P. Tiruveedhula, J. F. de Boer, and A. Roorda, "High-speed, image-based eye tracking with a scanning laser ophthalmoscope," *Biomedical Optics Express*, vol. 3, no. 10, pp. 2611, 2012.

[8] R. Engbert and R. Kliegl, "Microsaccades uncover the orientation of covert attention," *Vision Research*, vol. 43, no. 9, pp. 1035–1045, 2003.

[9] J. Otero-Millan, J. L. Castro, S. L. Macknik, and S. Martinez-Conde, "Unsupervised Clustering Method to detect microsaccades," *Journal of Vision*, vol. 14, no. 2, pp. 18–18, 2014.

[10] R. Zemblys, D. C. Niehorster, and K. Holmqvist, "Correction to: 'using machine learning to detect events in eye-tracking data," *Behavior Research Methods*, vol. 51, no. 1, pp. 451–452, 2018.

[11] S. Suthaharan, "Machine Learning Models and Algorithms for Big Data Classification: Thinking with Examples for Effective Learning," Springer US, Integrated Series in Information Systems, pp. 1-359, 2016.

[12] R. Joshi, and S. Suthaharan. "Pixel-Level Feature Space Modeling and Brain Tumor Detection Using Machine Learning," In 2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA), pp. 821-826. IEEE, 2020.

[13] K. K. Dansingani, K. K. Vupparaboina, S. T. Devarkonda, S. Jana, J. Chhablani, and K. B. Freund. "Amplitude-scan classification using artificial neural networks," Scientific reports 8, no. 1, pp. 1-7, 2018.

[14] A. Güven, and K. Sadık. "Classification of electro-oculogram signals using artificial neural network." Expert systems with applications 31, no. 1, pp. 199-205, 2006.

[15] S. Suthaharan. "Big data analytics: Machine learning and Bayesian learning perspectives—What is done? What is not?," Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 9, no. 1 (2019): e1283.

[16] F. Emmert-Streib, Z. Yang, H. Feng, S. Tripathi, and M. Dehmer, "An introductory review of Deep Learning for prediction models with Big Data," *Frontiers in Artificial Intelligence*, vol. 3, no. 4, pp. 1-23, 2020.

[17] B. T. Leonard, M. Zhang, V. Snyder, C. Holland, E. Bensinger, C. K. Sheehy, M. Collins, A. Kontos, E. A. Rossi, "Fixational Eye Movements Following Concussion," *Invest. Ophthalmol. Vis. Sci*, vol. 60, no. 9, pp. 1035, 2019.

[18] M. Zhang, E. Gofas-Salas, B. T. Leonard, Y. Rui, V. C. Snyder, H. M. Reecher, P. Mecê and E. A. Rossi. "Strip-based digital image registration for distortion minimization and robust eye motion measurement from scanned ophthalmic imaging systems." Biomedical Optics Express 12, no. 4, pp. 2353-2372, 2021.

[19] D. Jayachandra and A. Makur. "Directional variance: a measure to find the directionality in a given image segment," In Proceedings of 2010 IEEE International Symposium on Circuits and Systems, IEEE, pp. 1551-1554, 2010.

[20] M. Neuhaus and H. Bunke. "A graph matching based approach to fingerprint classification using directional variance," In International Conference on Audio-and Video-Based Biometric Person Authentication, Springer, Berlin, Heidelberg, pp. 191-200. 2005.

[21] A. P. Paplinski and J. F. Boyce. "Segmentation of a class of ophthalmological images using a directional variance operator and co-occurrence arrays." Optical Engineering 36, no. 11, pp. 3140-3147, 1997.