Eye Movement Classification with K-Means Clustering

Gentry Atkinson ([gma23@txstate.edu](mailto:gma23@txstate.edu))

|  |  |
| --- | --- |
| Research | 1 Hour |
| Code | 2 Hours |
| Refine/Debug | 4 Hours |
| Prepare Report/ Presentation | 5 Hours |

**Abstract:** Eye Tracking is increasingly an area of interest within HCI. It has been demonstrated to provide a more rapid and intuitive interface for many users than traditional user interfaces. However, the effectiveness of eye tracking platforms is often limited by their ability to distinguish fixations (wherein the eyes remain fixed on the neighborhood of a single point) and saccades (where the eye is in transition between fixations). Clustering is a well established within statistics and allows an analyst to programmatically distinguish between groupings within datasets based on a chosen distance measure. K-Means is one of the oldest and most trusted clustering algorithms. This paper demonstrates that K-Means can be effectively employed to classify fixations and saccades based on velocity features extracted from eye tracking data. The results of clustering based eye movement detection are comparable to traditional methods (such as I-VT) in accuracy but future refinements could promise substantial improvements in noise detection and hierarchical classification.

**1. Introduction-**

Eye tracking has been an area of focus for at least a hundred years in the field of human-machine interactions. Careful measurements of the eye movements of a subject can yield deep and meaningful insights into the intentions, desires, and physiology of a subject being monitored. As the performance of computing platforms continues to improve it is increasingly necessary to develop user interfaces that are responsive enough to allow users to take full advantage of the full performance of their machines.

Eye movement classification is one of the most fundamental and necessary tasks in the field of eye movement tracking. The human eye is capable of several modes of behavior including fixation, saccade, and smooth pursuit. Distinguishing between fixation and saccade is the most basic form of eye movement classification and is the simplest task that could be employed in order to test the viability of a classification technique.

There is a small body of work that demonstrates the viability of applying clustering algorithms to the task of biosignal processing. However of this corpus very little has been applied specifically to the task of eye movement classification. Clustering as a statistical technique has a decades long pedigree and has had demonstrable success in many disciplines of bioinformatics. Unsupervised machine learning is a very robust array of techniques that has many advantages to offer HCI.

Clustering techniques are able to automatically divide datasets in subsets on the bases of a distance measure acting on some features, or on the raw data. K-Means is one of the best established and most widely supported clustering algorithms. It is able to detect a specific number of clusters in data, whereas other algorithms return a number of cluster defined by the structure of the data or a variable number of clusters. This is well suited to the task of fixation vs. saccade classification where we would expect to find exactly to clusters.

In this paper K-Means is experimentally applied to the task of classifying fixations and saccades in a sample of 1800 data points collected from the eye movements of a subject at Texas State University in 2009. Feature extraction is limited to velocity and position in the x and y directions. The eye movements classifications given by K-Means are then compared to I-VT classification on the same feature set.

**2. Background-**

**2.1 K-Means** is a clustering algorithm which has been in common usage since 1967. It functions by guessing and then iteratively adjusting k cluster centroids within some data[1]. It is worth noting that K-Means does not try to balance the number of points in each cluster but rather finds centroids in the data the the points group around. K must be defined as an input parameter of the algorithm. Although many newer clustering algorithms have been defined, K-Means allows us to force the finding of exactly 2 clusters which is ideal for the separation of fixations and saccades. More complex algorithms might be better suited to more complex analysis, such as the detection of fixations, saccades, and smooth pursuits. K-Means also offers the advantage of running in quasi- O(n) time which is extremely fast by the standards of machine learning. Clustering algorithms are all very sensitive to the analyst's choice of distance measure. This experiment was conducted using a Euclidean distance measure. The K-Means algorithm is broadly described as:

|  |
| --- |
| Given set of data points s={s1,...,sn}  Randomly initialize set c={c1,...,ck} of k centroids  Initialize a set of results r={r1,...,rn} to zeros  Loop until c does not change:  for i = 1 to n:  min\_cluster = 0  min\_distance = MAX\_FLOAT  for j = 1 to k:  calculate distance(si, cj)  if (distance < min\_distance):  min\_cluster = j  min\_distance = distance  ri = min\_cluster  for i = 1 to k:  calculate the centroid of each cluster i  set ci = to centroid i    Return sets c and r. R is the cluster for each point in s, and c is the centroid of each cluster. |

Table 1: The K-Means Algorithm

**2.2 I-VT** classifies eye movements by comparing them to a fixed velocity threshold [2]. Points which fall below that threshold are marked as being part of a fixation while points exceeding the threshold are classified as saccades. I-VT is largely accepted as the simplest form of eye movement classification. Nonetheless, few of more modern classification techniques substantially outperform I-VT.

**2.3 Fixations and Saccades** are two of the six major eye movement types which include: fixations, saccades, smooth pursuit, optokinetic reflex, vestibulo-ocular reflex, and vergence [2]. Fixations are defined as the focus of a subject being held on a single point. Saccades by contrast are the rapid movements between fixations. Although there are many sub-types of saccades this work focuses on the broad classification of fixations and saccades.

**3. Methodology-**

**3.1 Feature Extraction** is first performed on the 1800 raw data points. A velocity in the x and y direction is calculated for each point and from them a combined velocity is calculated. The same feature set is used for both I-VT and for K-Means clustering. For K-Means it was necessary to strip out the position values to fit the data matrix required by the MatLab implementation of K-Means. Furthermore K-Means is given the absolute value of each velocity score rather than the raw value. This is to prevent the formation of three clusters in the data: one high velocity, one near zero, and one very negative.

**3.2 I-VT** compares each value in the extracted feature set with the defined velocity threshold. For this experiment a threshold value of >100 degrees per second marking a saccade and <30 degrees per second marking a fixation. The eye\_record data object is then updated with a classification of fixation, saccade, or noise for any points that did not pass either of the predefined thresholds.

**3.3 K-Means,** as discussed earlier, does not depend on defined thresholds. Rather, the velocity data from the eye\_record data object was separated into a 3x1800 matrix. The absolute value of the velocities were used rather than the raw values to prevent the formation of 3 clusters; highly positive, zero, and highly negative. For the purposes of eye movement classification highly negative velocity is the same as highly positive. The Matlab implementation of K-Means was then called on that matrix with a k value of 2 and Euclidean distances as the distance measure. This algorithm learns k centroids which are mean values for clusters within the data. The algorithm returns a matrix of 1800 clusters, which are the centroids closest to each data point. Points with missing velocity data were given a classification of noise.

**3.4 Scoring** was given to each classification in the form of a Fixation Quantitative Score, a Fixation Qualitative Score, and a Saccade Quantitative Score. All three scores rely on a comparison to the ground truth of the stimulus data. The two quantitative score compare the detected number of a classification to those present in the stimulus. In these cases a higher score is considered to be a better classification. The Fixation Qualitative Score summarizes the sum of errors between the stimulus data and the detected eye movements[2]. A lower Qualitative score is considered better[2].

**3.5 The Testing Platform** for these experiments is a consumer laptop running Matlab R2019a on an Intel I7 running 8 CPUs at 2.8 GHz.

**4. Results-**

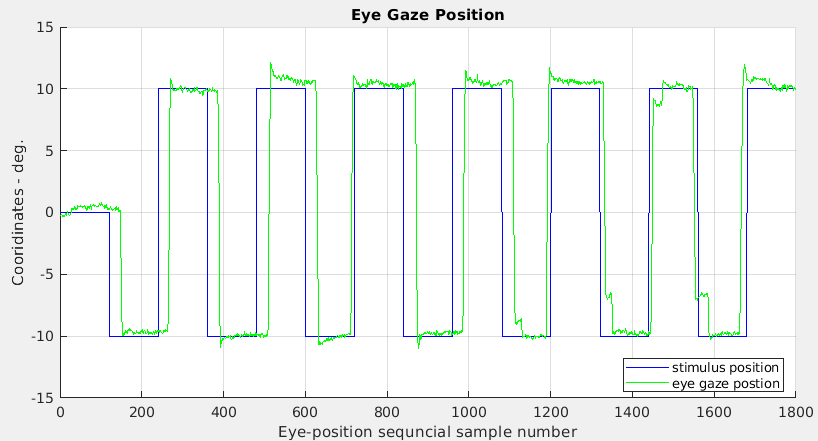
****

Figure 1: The stimulus data used as ground truth for all experiments.

Currently a comparison to ground truth is the only meaningful analyses of eye movement classification. This comes in two forms: comparison to the stimulus data and comparison to expert opinion.

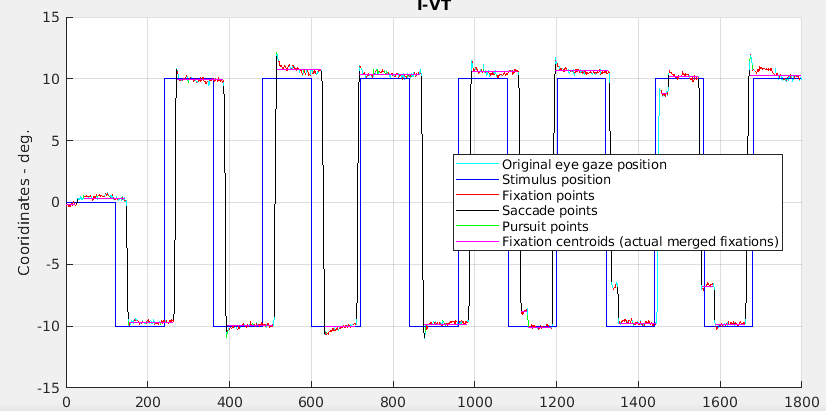


Figure 2: Classification performed by I-VT

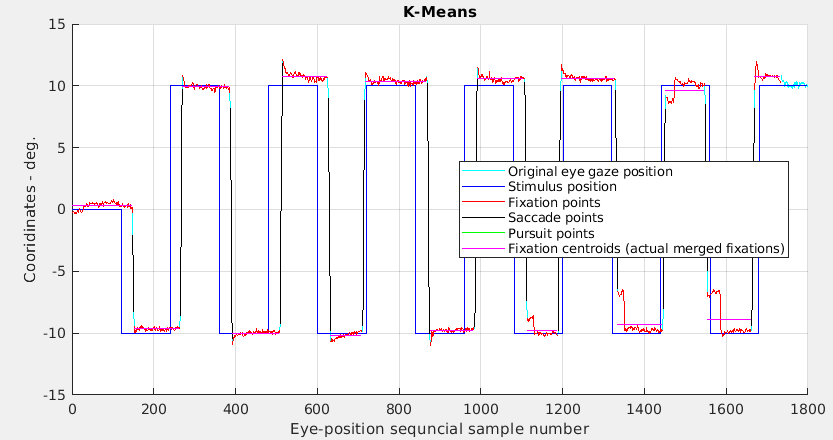


Figure 3: Classification performed by K-Means

Figures 2 and 3 present the eye movement classifications generated by I-VT and K-Means respectively. It is intuitively apparent that K-Means accomplishes more steady classification of the fixations but struggles to correctly identify the edge points on the saccades. We can also see the K-Means was able to detect a saccade around t=1400 that was entirely missed by I-VT.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Saccade Quantitative | Fixation Quantitative | Fixation Qualitative |
| I-VT | 94.8 | 66.67 | 0.31 |
| K-Means | 89.77 | 79.71 | 0.46 |

Table 2: Scoring of the two classification algorithms

Table 2 presents the Quantitative and Qualitative scores assessed for I-VT and K-Means. These values confirm that I-VT is doing a better job of detecting saccades within the eye tracking data while K-Means found a number of fixations that better matched the stimulus data. The qualitative scores for each algorithm were very similar with I-VT achieving a slightly lower (better) score.

**5. Conclusion and Discussion-**

These experiments demonstrate that K-Means is a viable eye movement classification algorithm. Although it is reasonable to conclude that I-VT has slightly outperformed K-Means on this data set it is important to remember that I-VT depends on a predefined velocity threshold and its performance will degrade substantially on data that does not so closely follow those threshold values.

It has been demonstrated that many conditions can affect the peak velocity of an eye moving through a fixade. Eye tracking data gathered from subjects who may be suffering from one of the pathologies would require careful tuning of the I-VT threshold in order for the algorithm to still classify the movements effectively. K-Means does not suffer from this shortcoming. Because K-Means learns it's divisions from the data it can be applied to any eye tracking dataset without the need to fine tune the algorithm. This kind of flexibility is important when classification is being done on bulk data or by an analyst who is not specifically familiar with the physiology of eye tracking.

Both methods of eye classification relied on the same data pre-processing steps. It is possible that with more accurate velocity calculations that the performance of either or both algorithms would improve substantially. In contrast to I-VT, K-Means can accept an arbitrary number of dimensions within the data points. The three dimensions provided to the algorithm (x velocity, y velocity, and combined velocity) were merely those which were being generated by the experimental setup. By providing additional data such as acceleration to the algorithm the accuracy could be greatly improved.

**6. Future Work-**

The performance of K-Means in these experiments has demonstrated the usefulness of clustering algorithms in the arena of eye movement classification. But K-Means is far from the only clustering algorithm. It was chosen for these experiments based on its simplicity and wide support. But there are many more modern and effective clustering algorithms available. Hierarchical algorithms such as Birch could provide far greater flexibility and detect more nuanced features like micro- and corrective-saccades. Density based algorithms like DBScan have proven very capable of noise filtering and could be useful for researchers using low cost, noisy eye trackers.

It will also be beneficial to explore a richer feature set with clustering algorithms. Velocity in 2 dimensions has proven to be sufficient for rudimentary eye movement classification but by adding additional features such as acceleration or local mean velocity, the performance of any clustering algorithm could be greatly improved. Related to this would be an exploration of different distance measures. Euclidean was employed in these experiments because it is well supported and was intuitively the correct choice. However, future experiments may reveal that another choice is better.

[1] McQueen J., (1967) Some methods for classification and analysis of multivariate observations, Proceedings of the 5th Berkeley Symposium on Math, Statistics, and Probability. Berkeley, CA

[2] Komogortsev O., Koh D., Jayarathna S., Gowda S., (2009) Qualitative and Quantitative Scoring and Evaluations of the Eye Movement Classification Algorithms, Technical Report: Texas State University. San Marcos, TX