NEURAL SPIKE SORTING AND CLASSIFICATION

ZAID H. BERJIS and AHMED K. AL-SULAIFANIE
Dept. Of Electrical and Computer Engineering, College of Engineering, University of Duhok,
Kurdistan Region-Iraq

ABSTRACT
Spike sorting is the process of separating the extracellular recording of the brain signal into one unit activity. There are a number of proposed algorithms for this purpose, but there is still no acceptable solution. In this paper a spike sorting method has been proposed based on the Euclidean distance of the most effective features of spikes represented by principle components (PCs) of the detected and aligned spikes. The assessments of the method, based on signal-to-noise ratio (SNR) representing background noise, showed that the method performed spike sorting to a high level of accuracy.

1. INTRODUCTION:

Neuroscientists depend mainly on the extracellular recordings of action potentials of neurons (also called spikes). These spikes are a means of communication between neural cells. Neurophysiologists may often want to know the timing of occurrence of these spikes and their corresponding shapes and most importantly to which neuron does each individual spike belong. The process that does the latter is called spike sorting. It is often a challenging process because of noise from the background and neurons in a local area have action potentials of similar shape and size [1]. Many algorithms have been suggested to do spike sorting efficiently [2][1][3].

In the early days of extracellular recordings only 3-5 neurons were being accessed by a single electrode [3] . To perform such activity the development of many devices has continued. The devices that have been developed so far are stereotrode [4], tetrode [5], and the multi-electrode array (MEA) [6]. Since the beginning of extracellular recordings there has been a desire to increase the number of recorded neurons by increasing the number of electrodes for several reasons.

A study [7] concluded that the number of neurons that must be accessed be between 8 and 10 neurons. Spike sorting algorithms must be developed to accommodate this increase in the vast amount of data being produced. At first, the spacing between electrodes were in magnitudes of $10^{(-4)}$ m such that each electrode can record from a single electrode [8] [9] [10]. So spike sorting of large amounts of data recorded by several electrodes is best processed for each electrode alone.

In more recent devices the spacing between electrodes are made much smaller than before in magnitudes of $10^{(-4)}$ m, so a certain spike from a one neuron can be detected by many electrodes in the vicinity of that neuron. In the same time, a single electrode can record the activity of many neurons in its neighborhood. This high density can help to assign spikes to their original neurons [5] [11] [12]); but signal from electrodes cannot be processed alone. Spike sorting algorithms should be adjusted to the new way of recording action potentials and the huge amount of data recorded. On the other hand when small numbers of electrodes were used like tetrodes, it was possible to use spike sorting methods of single electrodes and still get a very good performance [4] [13] [14] [15], this is not possible with the new designs of devices that are densely packed with hundreds of electrodes. Complementary Metal-Oxide Semiconductor CMOS devices with electrodes exceeding the thousand mark have been tried and are often used [16] [17] [18] [19], requiring new algorithms of spike sorting specially designed for this number of electrodes.

zaid.berjis@gmail.com

In this paper, a method is proposed for spike sorting based on the Euclidean distance of a scatter plot of the most effective features extracted by the principle components of the spikes.

This paper is divided into the following sections, section 1 is an introduction to spike sorting. In section 2, spike sorting is presented with the main steps involved in the process. In section 3, a powerful tool is laid out in a new way and detailed steps that is the principle component analysis (PCA). In section 4, the proposed method is explained. In section 5, the simulated results are shown with graphs and performance assessment was done as well to determine the accuracy of the method. In section 6, the results and performance have been discussed. Lastly, in section 7, the conclusion of the work is given.

2. SPIKE SORTING

The first step in spike sorting method involves the Detection and Alignment of Spikes. Amplitude thresholding is a popular method for detecting spikes. Choosing the right threshold is a difficult decision because a compromise between a high threshold and a low threshold must be made to avoid false positives when a low threshold is used and false negatives when a high threshold is used. A manual threshold can be set as the case in online spike detection systems. Automatic thresholding is more desirable especially when the number of channels used is very large. Multiples of the standard deviation of the recordings (spikes included) is taken for automatic thresholding [20]. But, using the standard deviation alone does not work as desired in cases of high firing rates and high amplitudes. On the other hand, if we use an estimation that uses the median as a base as given in (1), the effect of spikes on the threshold will be reduced significantly.

$$\sigma_n = median\{|x|/0.6745\} \qquad (1)$$

Where x is the bandpass filtered signal and σ_n is an estimate of the standard deviation of the background noise [21].

The threshold that has been adopted (Thr) automatically set to [22]:

$$Thr = 4\sigma_n \qquad (2)$$

After the detection of spikes, they must be stored to be clustered.

Regarding the storage of spikes there are two issues that must be discussed briefly. The first of

which is how many datapoints must be stored. It depends on the sampling frequency usually it is desired to store the whole spike ; e.g. a 2 ms of data with sampling frequency of 30 kHz requires the storage of 60 datapoints. Note that there are methods for feature extraction like wavelets that need the number of datapoints to be a power of 2. To apply this for the above-mentioned example, 64 points need to be stored for optimal spike representation [23].

The second issue is about the alignment of these spikes, which can be aligned to their maximum. But, because of insufficient sampling the maximum can occur at different points of the spike shape. To avoid this problem that can lead to over-clustering, interpolation of waveforms can be used to oversample the spike shapes, for example using cubic splines. Then alignment can be done easily after the interpolation and later can be decimated to the original sampling frequency [24].

3. FEATURE EXTRACTION USING PRINCIPAL COMPONENT ANALYSIS (PCA)

After the alignment of spikes, a decision must be made to how many clusters of neurons the obtained spikes can be classified. This problem can be reliably resolved by comparison of spikes to one another. In reality, the spike sorting process is a formidable task due to the noise present in the background, the spike waveforms variability, distinguishing between spikes is also because they are similar to a high degree to each other. Mathematically speaking, every spike lies in an m-dimensional space as a single point. The m is usually very large, the curse of dimensionality is the main obstacle for clustering of spikes in such a multidimensional space. So a certain method is needed to reduce the dimension of the space representing the spikes.

PCA is a technique for choosing the main features of spikes automatically. The main purpose behind PCA is to obtain an ordered set of orthogonal basis vectors that represent the directions in the data of largest variation. The data being the spikes obtained from the recorded signals. Each signal is centered in the spike's maximum to reduce the variability of the spike shapes. To represent any spike the principal components are multiplied by a scale and added together. The scale factor for each component is

167

zaid.berjis@gmail.com

sometimes called the score. The ith score is calculated by [1]

$$s_i = \sum_t c_i(t)x(t) \qquad (3)$$

Where $x(t)$ is the spike and $c_i(t)$ is the ith principal component.

Because the components are ordered from the highest variability to the lowest, adding together the first k components will describe the most variation in the data. Adding more components results in small corrections until the exact spike is described. The principal component vectors are obtained by computing the eigenvectors of the covariance matrix of the data.
Let X be a $M \times N$ matrix,

$$X = \begin{bmatrix} x_{11} & x_{12} & x_{13} & & x_{1N} \\ x_{21} & x_{22} & x_{23} & \cdots & x_{11} \\ \vdots & \vdots & \vdots & & \vdots \\ x_{M1} & x_{M2} & x_{M3} & & x_{MN} \end{bmatrix} \qquad (4)$$

Where each row is an individual spike consisting of N samples. Next the mean of $\bar{x}$ must be calculated, so we obtain a mean matrix $\bar{x}$

$$\bar{x} = \begin{bmatrix} \bar{x}_1 \\ \bar{x}_2 \\ \bar{x}_3 \\ \vdots \\ \bar{x}_m \end{bmatrix} \qquad (5)$$

and subtract the mean from each element in that row to produce the demeaned matrix B.

$$B = X - \bar{x} \qquad (6)$$

Then find the covariance matrix C of matrix B as follows:

$$C = \frac{1}{N-1}BB^T \qquad (7)$$

The diagonal of matrix C is the variance of B and the off-diagonal elements are the covariances. The goal of PCA is to make the

covariance vanish, which means remove the correlation. The main diagonal of C contains the most important information about the variables of the demeaned matrix B, the covariances are considered to be the redundancy in the data. If there is a zero covariance or zero correlation then $C(i,j)$, for $i \neq j$, is zero. It may be clear that analysis of multivariate data would be simpler when the signals are uncorrelated that is a covariance matrix is diagonal which means that all off-diagonal elements are zero and this is exactly the goal of PCA. Note that the correlation between two variables x and y is a normalized version of the covariance between them $\rho_{xy} = Cov(x,y)/\sigma_x\sigma_y$ where $\sigma_x \ and \ \sigma_y$ are the standard deviations of x and y respectively [26].

Now the orthogonal eigenvectors of the covariance C is calculated which returns the principal components PC of the spike's matrix X.

$$PC = \begin{bmatrix} c_{11} & c_{12} & c_{13} & \cdots & c_{1n} \\ c_{21} & c_{22} & c_{23} & \cdots & c_{2n} \\ c_{31} & c_{32} & c_{33} & \cdots & c_{3n} \\ & & \vdots & & \\ c_{n1} & c_{n2} & c_{n3} & \cdots & c_{nn} \end{bmatrix} \qquad (8)$$

Where each column of PC contains coefficients for one principal component, and the columns are in a descending order of component variance. The principal component scores S are representations of matrix X in the principal component space and estimated by projection of X on PC as follows:

$$S = B\,PC \qquad (9)$$

Then each spike X_i can be represented as a weighted sum of the principal component vectors with the corresponding weights or scale factors, so called *score coefficient* $s(i,j)$, which are evaluated by the inner product. Therefore the jth score coefficient $s(i,j)$ can be evaluated by the scalar product.

$$s(i,j) = \langle B_i, PC_j \rangle = \sum_{n=1}^{N} B(i,n) \times c(n,j) \qquad (10)$$

zaid.berjis@gmail.com

We can say that projection of waveform X_i on the principal component PC_j is a single

one feature of X_i extracted from PC_j. Then the scores $s(i,j)$, can be considered as features describing the ith spike waveform X_i. The scores of the first and second components can serve as features for classifying different spikes.

coefficient factor (i,j), and this is considered as the

$$[s(:,1)s(:,2)] = \begin{bmatrix} s(1,1)\ s(1,2) \\ s(2,1)\ s(2,2) \\ s(3,1)\ s(3,2) \\ \vdots\qquad \vdots \\ s(M,1)s(M,2) \end{bmatrix} = \begin{bmatrix} feature\ 1\ of\ x_1 & feature\ 2\ of\ x_1 \\ feature\ 1\ of\ x_2 & feature\ 2\ of\ x_2 \\ feature\ 1\ of\ x_3 & feature\ 2\ of\ x_3 \\ \vdots & \vdots \\ feature\ 1\ of\ x_M & feature\ 2\ of\ x_M \end{bmatrix} (11)$$

F
igure 3 shows a scatter plot of these two scores of $M = 2948$ waveforms each with $N = 64$ data points (samples).

It is desired sometimes to reconstruct the original waveform X from the score matrix and the principal components. It could be done as follows. Perform matrix multiplication between S and PC^T.

$$B = S\ PC^T \qquad (12)$$
$$X - \bar{x} = S\ PC^T \qquad (13)$$

$$X = S\ PC^T + \bar{x} \qquad (14)$$

4. THE PROPOSED METHOD OF SPIKE SORTING

Our focus in this paper is on spike sorting so we will start our experimentation on the raw data as shown in $figure$ 1 The filtered, detected, and aligned data looks like the one shown in $figure$ 2 .
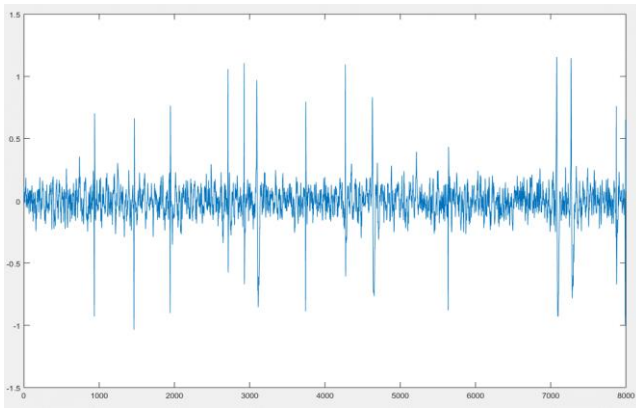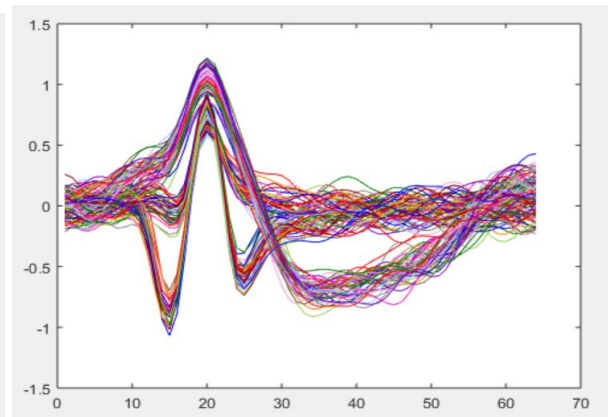

Fig. (1): Raw data

Fig. (2): Filtered, detected, and aligned spikes.

Our proposed method entails the following steps; first we calculate the principal components for the aligned spikes for dimensional reduction. Then the first and second principal components of the data were plotted

against each other, the first principal component on the horizontal axis and the second principal component on the vertical axis, as shown in $figure$ 3 .
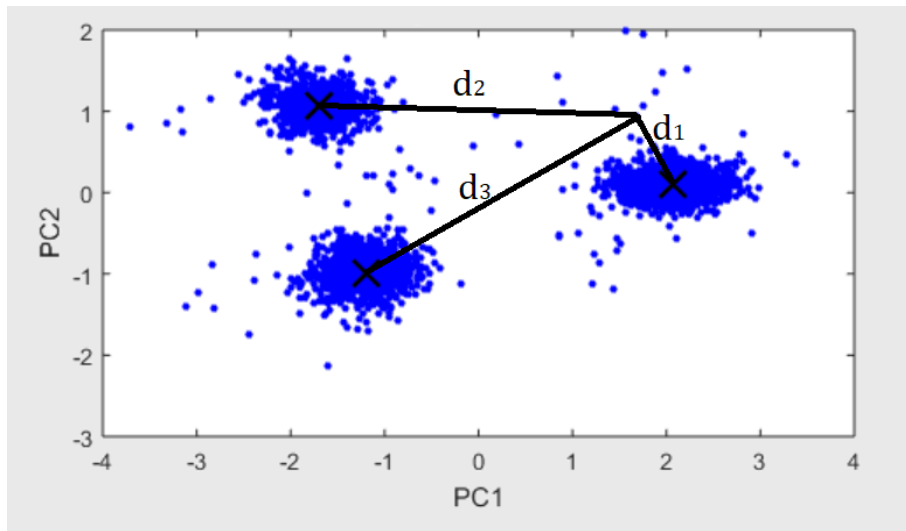
Fig. (3): PC1 Vs. PC2 graph of the data with an example of how the distance should
be measured from a point to the three centroids of the clusters

Then the Euclidean distance was calculated from the centroid of each cluster to every point in figure 3. Now each point has three distances d1, d2, and d3 from the centroid. The shortest of which can be inferred to be the correct cluster of that point.

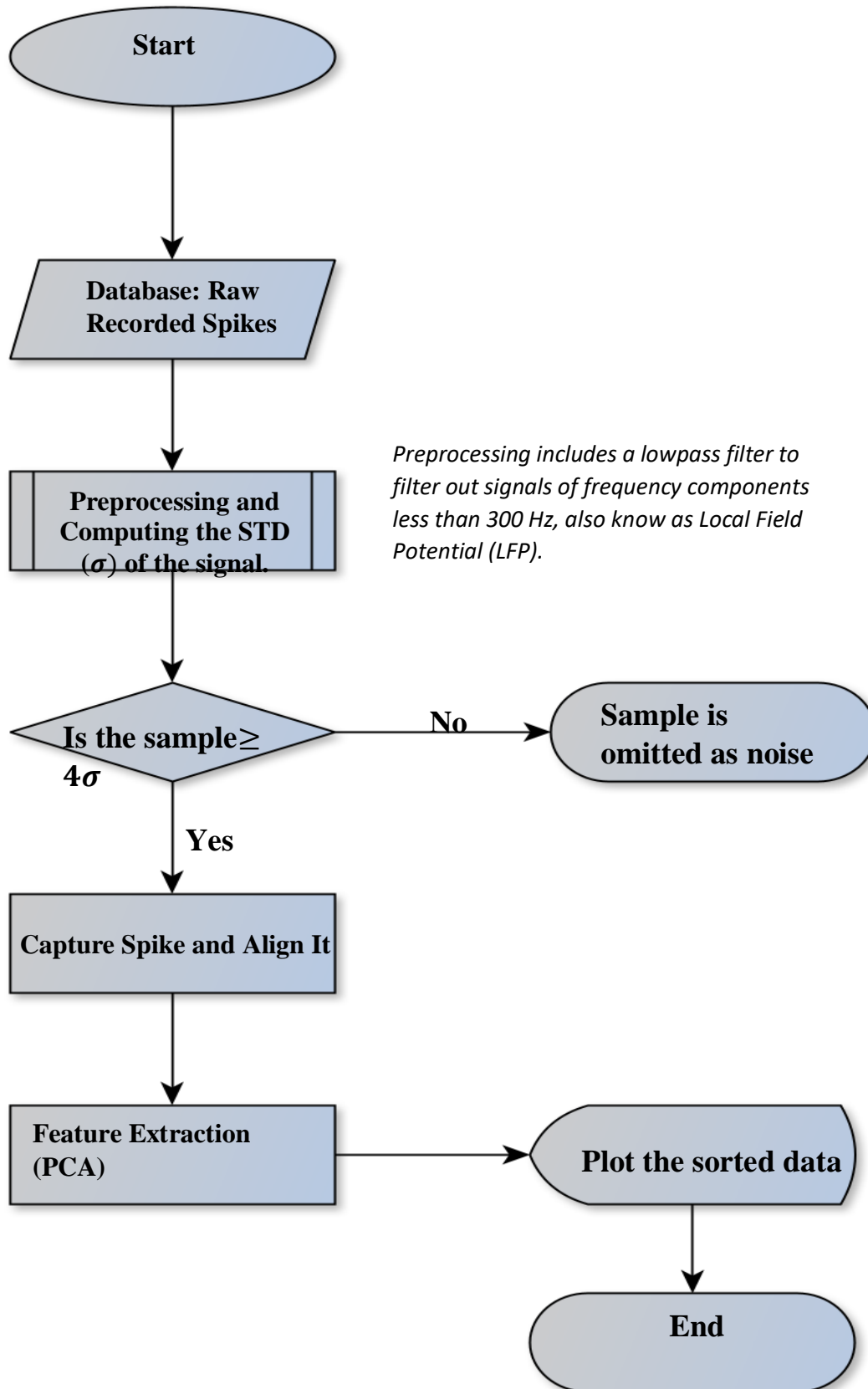A flow chart of the entire process is shown in figure 4.

zaid.berjis@gmail.com

Start

Database: Raw Recorded Spikes

Preprocessing and Computing the STD (σ) of the signal.

Preprocessing includes a lowpass filter to filter out signals of frequency components less than 300 Hz, also know as Local Field Potential (LFP).

Is the sample$\geq$ 4σ

No → **Sample is omitted as noise**

Yes

Capture Spike and Align It

Feature Extraction (PCA) → **Plot the sorted data**

End

Fig. (4): A flowchart showing the main steps required to perform the spike sorting process

5. SIMULATION RESULT OF SPIKE SORTING

171

zaid.berjis@gmail.com

Using synthetic data is better, in quantifying the performance of spike sorting methods, than using real data because there is no ground proof that the sorted data are correctly clustered with their original groups; i.e. a real spike will be assigned to a certain neuron and still it will not be clear whether this is correct or not. There are some exceptions though when the spikes are recorded extracellularly and intracellularly then their shapes and timing are known because of the correlation between the two recordings; more specifically the extracellular recording is the differentiation of the extracellular recording.[13].

5.1 Simulation Data:

We used three different spikes of two sets of data, shown in $figure$ 5. These spikes are extracted from the data publicly available online at [https://www2.le.ac.uk/centres/csn/software, provided by Quiroga et al].These three spikes where numbered 1,2, and 3 for tracking purposes.
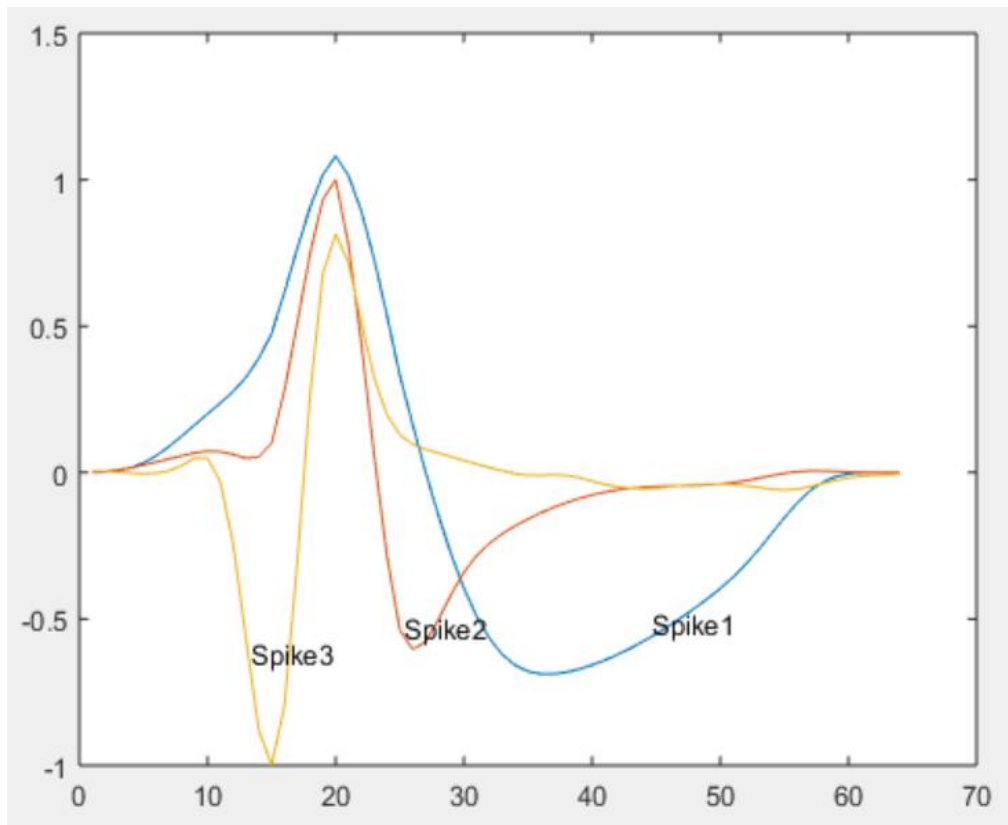

Fig. (5): Three spikes numbered 1,2, and 3. Amplitude Vs. 64 Bits

A poisson distribution sequence was generated, see $figure$ 6, to reproduce a set of spikes consisting of these three spikes only.
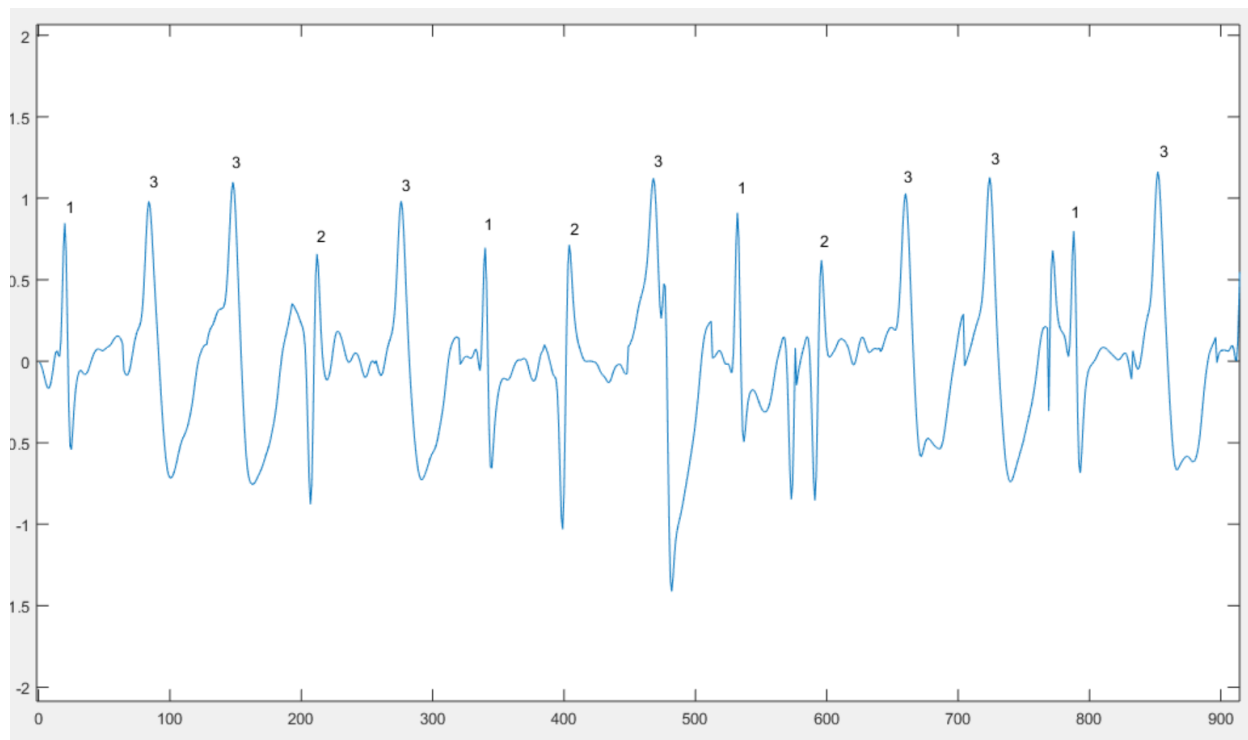
172

zaid.berjis@gmail.com

Fig. (6) : 900 ms of data showing the sequence generated using Poisson distribution

Then we generated noise that has normal distribution characteristics and a unity standard deviation as shown in $figure$ 7.
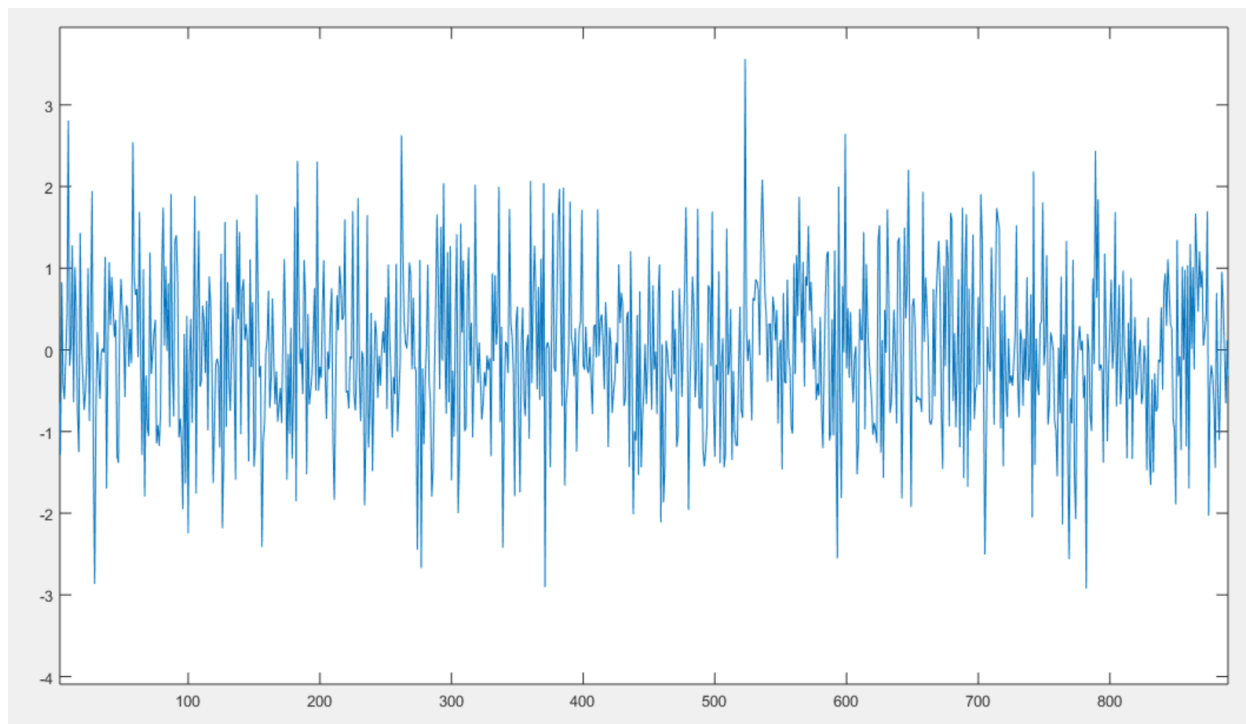

Fig. (7): Gaussian noise with unity standard deviation

zaid.berjis@gmail.com

Then the aligned spikes, that have been detected earlier, were contaminated by the generated noise.

5.2 Spike Feature Extraction:
As mentioned previously that feature extraction is an essential step in spike sorting routine. There are several feature extraction methods that can be used. Here a very powerful and popular one is used, which is the principal component analysis or PCA using the MATLAB function pca(data). We obtained the graph shown in $figure\ 8$.
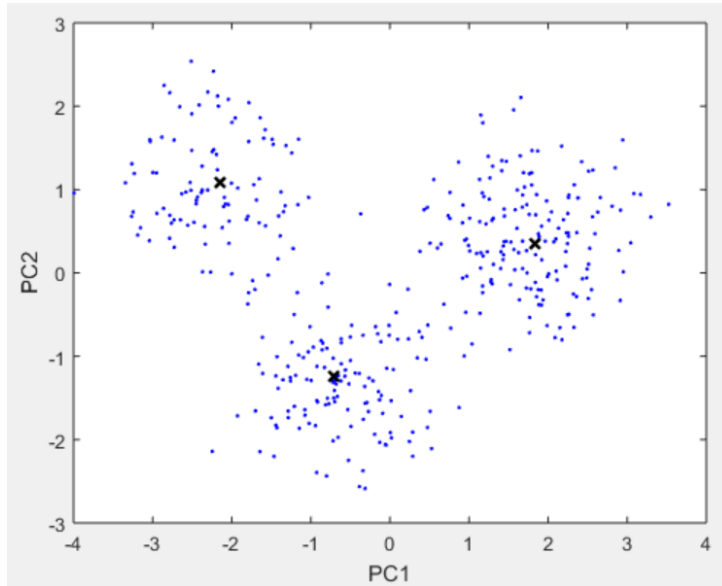

Fig. (8): PC1 Vs. PC2 of the data used in the study

5.3 Sorting:
The clustering method depends on the shortest Euclidean distance to the centers of the observed clusters. To decide how many clusters we have in the data, we look at the graph of the PC1 Vs. PC2 shown in $figure\ 8$. Clearly, we have three distinct clusters. The centroids will be chosen using the PC1 Vs. PC2 of the data before adding the noise for simulation purposes only. Or we could use the k-means algorithm to find the centroids of these clusters. Now each point will have three distances measured to each of these centroids. The shortest indicates that the point belongs to the cluster of that centroid.

5.4 Accuracy and Performance:
Determining the accuracy of spike sorting is essential for the validation of the method of spike sorting. We will vary the noise level represented by the Signal-to-Noise Ratio (SNR). When a spike is correctly grouped with its actual cluster, it is called true positive (TP), and when a spike is incorrectly grouped, it is called false positive (FP). The accuracy is defined as the ratio of TP to the total number of spikes [26].

$$Accuracy = TP / Total\ \#Spikes \qquad (15)$$

Different number of spikes where tested 50,150,250, and 350 spikes as shown in $figure\ 9$ below. We found that the method performs better when the number of spikes is high even for low SNR and that is well suited with the real data since each recording session takes in thousands of spikes. Taking small number of spikes is not practical because of the inconvenience caused for the patient. Note that the SNR is defined as ratio of the peak-to-peak voltage to twice the standard deviation of the spike.

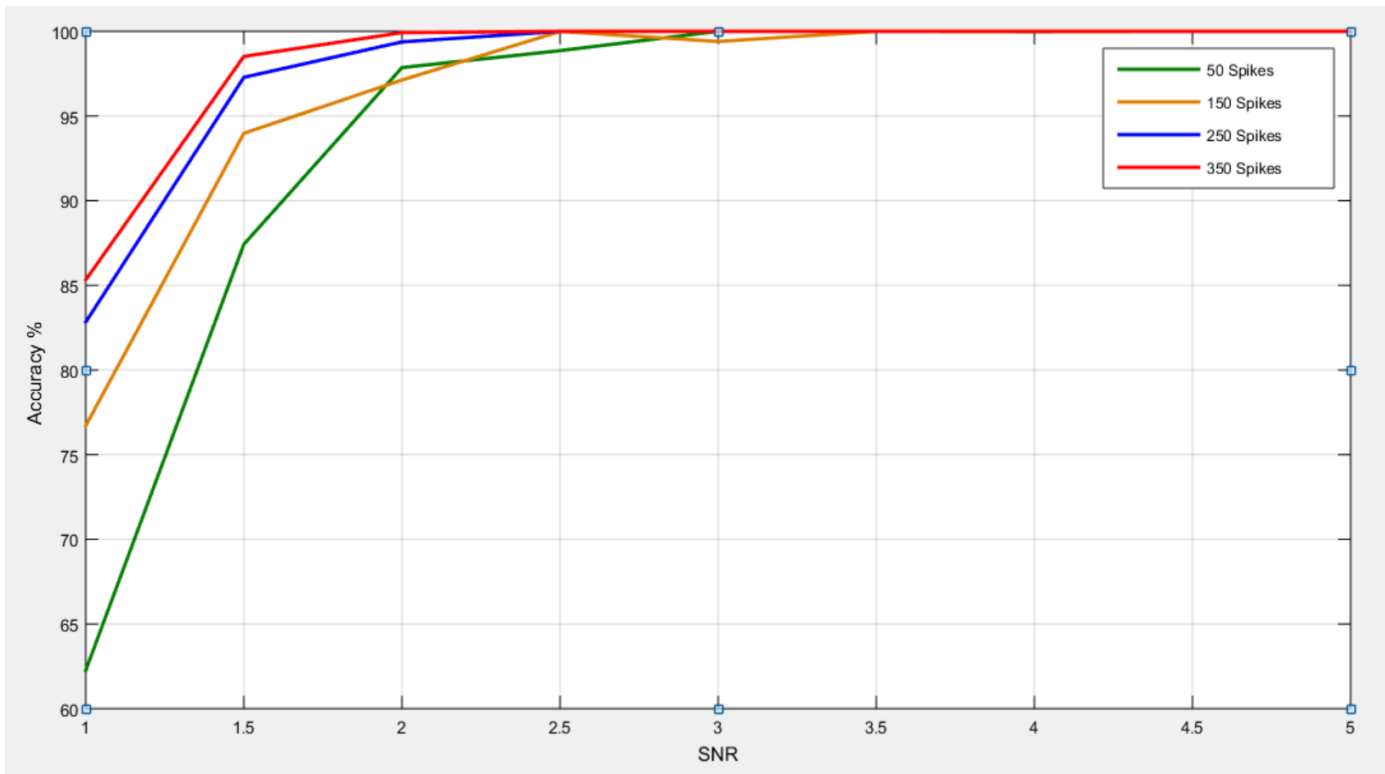$$SNR = \frac{V_{PP}}{2\sigma_n} \qquad (16)$$

174

Fig. (9): Accuracy Vs SNR for different number of spikes.

6. DISCUSSION

In this study, spikes were generated using poisson distribution, and then Gaussian noise was added to the spikes with different noise levels represented by the SNR. Now the PCA was used as the method of feature extraction which is a necessary step in spike sorting because it reduces the number of dimensions of each spike as shown in $figure$ 10 and 11. Then k distances were measured for every point in this new space. Point A in $figure$ 10 (a) has three [because we can see that there are three clusters] distances d1, d2, and d3 as shown below, d3 is the shortest distance so point A must be clustered with cluster 3 and so on.

To assess the performance of this method different numbers of spikes were sorted having different levels of noise (SNR). The method is far more useful when the number of spikes is high even with low SNR values. Typically, a data with an $SNR \geq 5$ can be sorted correctly with a high level of confidence [22].
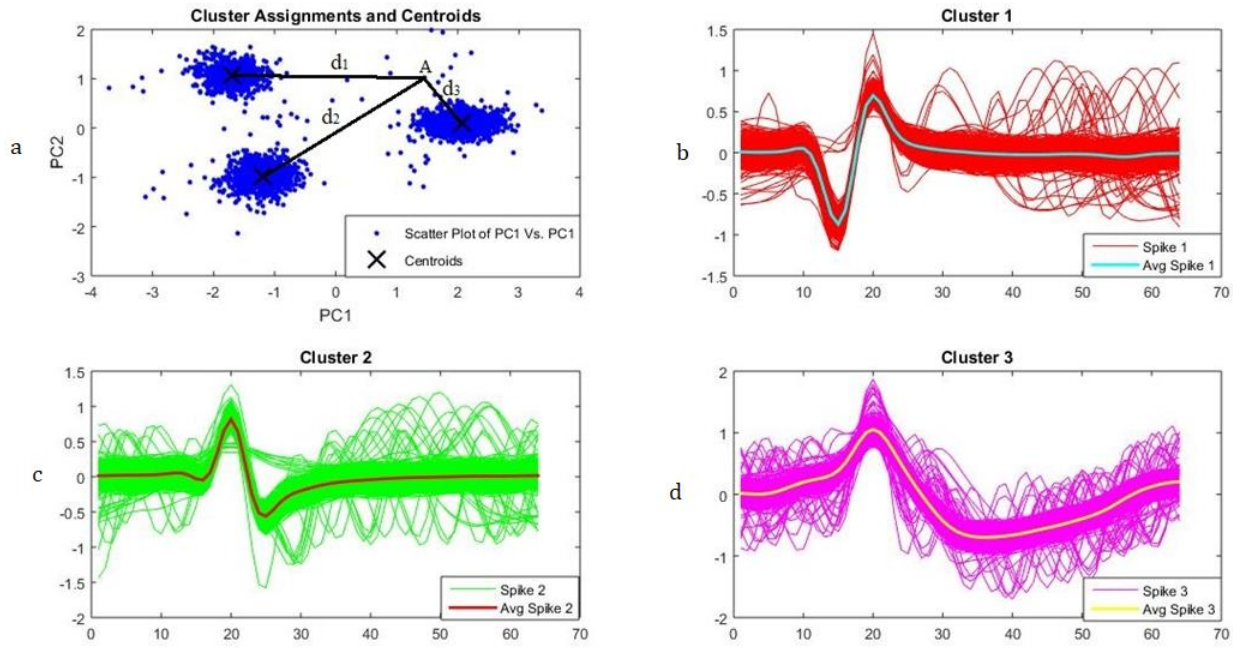
175

zaid.berjis@gmail.com

Fig. (10: a) Scatter plot of PC1 Vs. PC2 b) Spikes for cluster 1 c) Spikes for cluster 2 d) Spikes for cluster 3
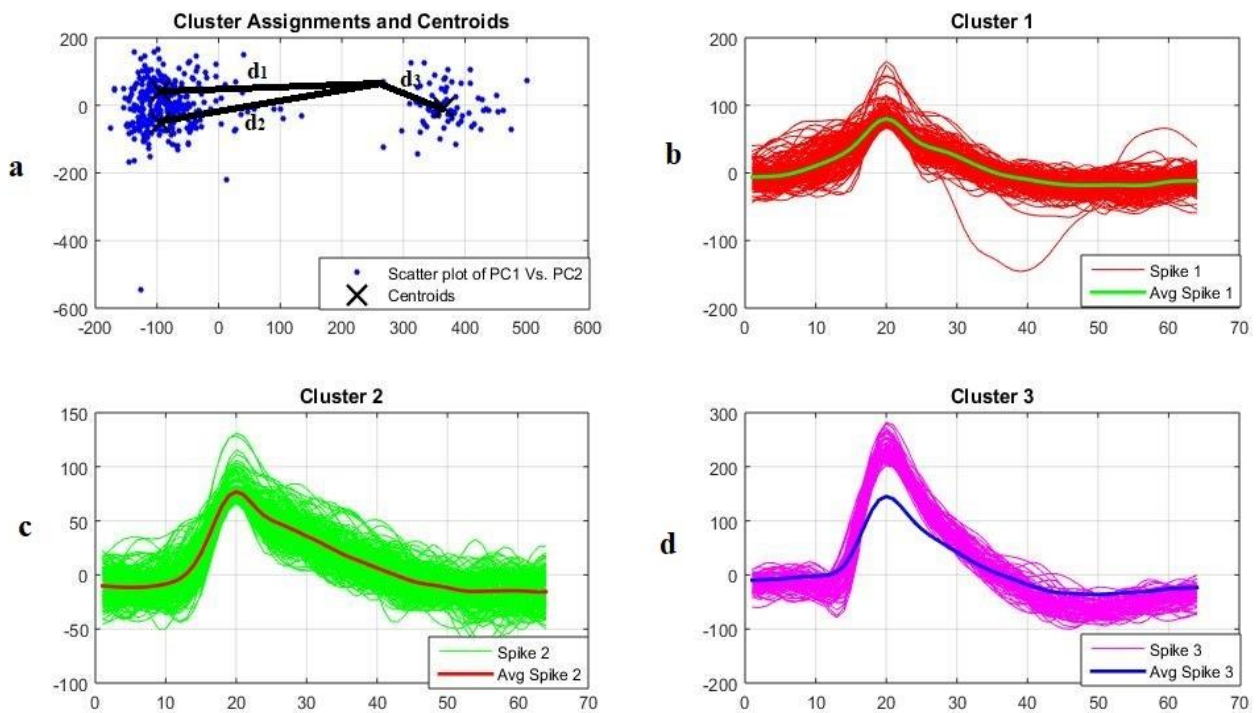

Fig. 11: a) Scatter plot of PC1 Vs. PC2 b) Spikes for cluster 1 c) Spikes for cluster 2 d) Spikes for cluster 3

A different set of data is used here to verify the validity of the method used

176

7. CONCLUSION

Spike sorting can be divided into three main steps: Spike detection and Alignment, Feature extraction, and Clustering. The feature extraction is an essential step in spike sorting routine. Feature extraction emphasizes the difference between waveforms and reduces the dimensionality of the data. In this paper, a method for spike sorting was presented with a quantitative measure for the evaluation of this method. The proposed method used simulated data and proved to perform better with big data rather than small data. The spikes were generated using poisson distribution, and then Gaussian noise was added to the spikes with different noise levels represented by the SNR. We used the PCA features to represent the data with minimal dimensions and reduction in noise effect.

REFERENCES

Lewicki, M. S. (1998). A review of methods for spike sorting: the detection and classification of neural action potentials. *Network: Computation in Neural Systems*, R53-R78.

Rey, H., Ison, M., Pedreira, C., Antonio Valentin, A., Gonzalo, Selway, R., Quiroga, Q. R. (2015). Single-cell recordings in the human medial temporal lobe. *Journal of Anatomy*.

Gerstein, G. L., & Clark, W. A. (1964). Simultaneous Studies of Firing Patterns in Several Neurons. *Science*, 1325-1327.

McNaughton, B., Barnes, C., & O'Keefe, J. (1983). The contributions of position, direction, and velocity to single unit activity in the hippocampus of freely-moving rats. *Experimental Brain Research*, Volume 52, Issue 1, pp 41–49.

Gray CM., Maldonado PW., Wilson M., and McNaughton B. (1995). Tetrodes markedly improve the reliability and yield of multiple single-unit isolation from multi-unit recordings in cat striate cortex. J Neurosci Methods.; 63(1-2):43-54.

Thomas Jr. CA., Springer P., Loeb GE., Berwald-Netter Y., and Okun LM. (1972). A miniature microelectrode array to monitor the bioelectric activity of cultured cells. Exp Cell Res.;74(1):61-6.

Pedreira, C., Martinez, J., J.Ison, M., & Quiroga, R. Q. (2012). How many neurons can we see with current spike sorting algorithms? *Journal of Neuroscience Methods*, Volume 211, Issue 1, Pages 58-65.

Jones, K. E., Campbell, P. K., & Normann, R. A. (1992). A Glass/Silicon Composite Intracortical Electrode Array. *Annals of Biomedical Engineering*, 20, 423-437.

Shoham, S., Fellows, M. R., & Normann, R. A. (2003). Robust, automatic spike sorting using mixtures of multivariate t-distributions. *Journal of Neuroscience Methods*.

Carlson, D., & Carin, L. (2019). Continuing progress of spike sorting in the era of big data. *Current Opinion in Neurobiology*, Volume 55, Pages 90-96.

Franke, F., Pröpper, R., Alle, H., Meier, P., Geiger, J., Obermayer, K., & Munk, M. (2015). Spike sorting of synchronous spikes from local neuron ensembles. *Journal of Neuroscience Methods*, 114(4):2535-49.

Trautmann, E., Sergey, D., Subhaneil, L., Ames, K., T.Kaufman, M., J.O'Shea, D., . . . V.Shenoy, K. (2019). Accurate Estimation of Neural Population Dynamics without Spike Sorting. *Neuron*, Volume 103, Issue 2, Pages 292-308.

Harris, K., Henze, D., Csicsvari, J., Hirase, H., & Buzsáki, G. (2000). Accuracy of tetrode spike separation as determined by simultaneous intracellular and extracellular measurements. *Journal Of Neurophysiology*, 84(1):401-14.

Gao, Solages, C., & Lena, C. (2012). Tetrode recordings in the cerebellar cortex. *Journal of Physiology-Paris*, 106(3-4):128-36.

Caro-Martín, C. R., Delgado-García, J. M., Gruart, A., & Sánchez-Campusano, R. (2018). Spike sorting based on shape, phase, and distribution features, and K-TOPS clustering with validity and error indices. *Scientific Reports*.

Berdondinia, L., Wala, P. v., Guenata, O., Rooija, N. d., & Koudelka-Hepa, M. (2005). High-density electrode array for imaging in vitro

177

electrophysiological activity. *Biosensors and Bioelectronics*, 167-174.

Fiscella, M., Farrow, K., Jones, I. L., Jäckel, D., Müller, J., Frey, U., . . . Hierlemann, A. (2012). Recording from defined populations of retinal ganglion cells using a high-density CMOS-integrated microelectrode array with real-time switchable electrode selection. *Journal of Neuroscience Methods*, 103-113.

[18] Müller, J., Ballini, M., Livi, P., Chen, Y., Radivojevic, M., Shadmani, A., . . . Hierlemann, A. (2015). High-resolution CMOS MEA platform to study neurons at subcellular, cellular, and network levels. *Lab on a Chip*, 15(13):2767-80.

Hilgen, G., M, S., S, P., JO, M., IE, K., S, U., . . . MH, H. (2017). Unsupervised Spike Sorting for Large-Scale, High-Density Multielectrode Arrays. *Cell Reports*, 2521–2532.

Pouzat, C., Mazor, O., & Laurent, G. (2002). Using noise signature to optimize spike-sorting and to assess neuronal classification quality. *Journal of Neuroscience Methods*, 122(1):43-57.

.

Donoho, D. L. (1995). De-noising by soft-thresholding. *IEEE Transactions on Information Theory*, 613 - 627.

Quiroga, R. Q., Nadasdy, Z., & Ben-Shaul, Y. (2004). Unsupervised Spike Detection and Sorting with Wavelets and Superparamagnetic Clustering. *Neural Computation*, 1661-1687.

Nenadic, Z., & Burdick, J. (2005). Spike detection using the continuous wavelet transform. IEEE Trans Biomed Eng., 74-87.

Quiroga, R. Q., Reddy, L., Koch, C., & Fried, I. (2007). Decoding visual inputs from multiple neurons in the human temporal lobe. Journal of Neurophysiology, 1997-2007.

Drongelen, W. v. (2018). Signal Processing for Neuroscientists . In W. v. Drongelen, *Signal Processing for Neuroscientists* (pp. 126-131). London: Academic Press.

Yang, Y., Boling, C. S., Kamboh, A. M., & Mason, A. J. (2015). Adaptive Threshold Neural Spike Detector Using Stationary Wavelet Transform in CMOS. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 946 - 955

zaid.berjis@gmail.com