Seizure Detection Using Gilbert's Algorithm

Abdel-Malik M. Sabreen*, Adel A. Samir*, Lojaine A. ElMahdy*, Mirna H. Ibrahim*, Mohamed H. Tawfik*,
Omneia O. ElShaer* and Hassan Mostafa*‡
*Department of Electronics and Communications Engineering, Cairo University, Egypt

***Abstract*—Seizure detection for epileptic patients can be done using Support Vector Machines (SVMs). SVMs are a well-established method in classification between seizure and non-seizure points. One of the SVM trainers is Gilbert's Algorithm. This paper elaborates Gilbert's Algorithm role in training SVM to succeed in performing seizure detection. FPGA is used to accelerate the SVM training because of its reconfigurability. The reached results are highlighted and discussed as well as the used power and resources.**

***Index Terms*—SVM,Gilbert's Algorithm, Seizure detection, Epilepsy,FPGA**

I. Introduction

Epilepsy is a condition in which the patient has recurrent seizures, it is a neurological disorder caused by abnormal electrical discharges in the brain. Approximately 0.7% of the world population suffers from epileptic seizures. More than 50 million people are affected by Epilepsy worldwide with more than 2.2 million in the US [1].

The normal operation of the functional region of the brain cortex is the depolarization of the cell, where the voltage increases because of the entry of the sodium. Then, repolarization occurs by the opening of the potassium channel until it reaches the complete repolarization and the cell becomes in its normal state again [2]. In seizure, the depolarization occurs, but the repolarization doesn't, which keeps the cell in its abnormal state due to the lack of inhibitory neurotransmitters. A solution to the problem is to build a chip that will be implanted in the patient's brain. The target of this chip is to employ Machine Learning (ML) algorithms to be able to detect a seizure for any patient and hence trigger electric stimuli that will be able to retrieve the brain back to its normal conditions.

Machine learning could be considered as a method of data analysis that automates analytical model building. It proves its importance in efficient solving for a classification problem for the flexibility and adaptivity of the classification parameters. The electrical activities of patients are recorded using Electroencephalogram (EEG) which measures the voltage fluctuations of the brain. A classification method is needed to categorize between seizures and non-seizures. Feature extraction and decision layers are used to first build an effective Machine Learning algorithm that is able to efficiently detect a seizure from the readings of the EEG signal using [MATLAB] tool. Then, this algorithm will be mapped to low level hardware descriptive language to be able to define its hardware layout and simulation. One of the machine learning algorithms is the support vector machine (SVM) used to obtain maximum margin around a hyperplane needed for classification process. Maximum margin defines the maximum distance between the hyperplane and the closest point to the plane which defines the support vector (Xn)[3].

SVM is a hyperplane that separates a set of positive examples from a set of negative examples with maximum margin (see Figure 1). In the linear case, the margin is defined by the distance of the hyperplane to the nearest of the positive and negative examples[4].

The use of FPGA device would effectively aid the online training of the SVM especially as it is considered a large-scale classification problem; hence, taking advantage of the FPGA resources maximizing its utilization would be very helpful for the design problem. In this work, a seizure detection algorithm using Gilbert's algorithm has been chosen to be researched, implemented and analyzed[5].

There are different types of SVM trainers. The popular trainers include Sequential Minimal Optimization (SMO)[6], Gradient Descent[7], [8]and Gilbert's Algorithm. It is noteworthy to say that Gilbert's algorithm was chosen to be implemented in this paper as the SVM trainer.

Gilbert's algorithm has some features over SMO and Nearest point algorithm as it's simpler than SMO and NPA. Gilbert's CPU run time is much faster than both of them. Beside, it uses fewer support vectors for SVM[9].

II. Gilbert's Algorithm

In 1966, Gilbert's algorithm was introduced to the machine learning field. In 2000, it has been used as a support vector machine but it had slow convergence time. In 2005, Gilbert's had been modified to solve this slow rate of convergence. Recently, various research works approach the SVM training from a geometric view of the problem. These proposed methods consider Gilbert's Algorithm to the geometric expression of the SVM training problem.

In [10] Gilbert's Algorithm is discussed in details. However in this paper, Gilbert's algorithm is introduced briefly. Gilbert's Algorithm uses the concept of Minkowski difference. Given two convex hulls representing the two classes of positive and negative seizures, the normal to the separating hyperplane $2/(\|w^*\|)$ can be obtained by $\|u^*-v^*\|$ as both points u^* and v^* are the closest points on the two convex hull of the two classes as shown in figure (1).

The problem of finding the minimum distance between two convex hulls is known as the nearest point problem (NPP). Gilbert's algorithm is one of the first algorithms suggested for solving NPP. It is applied on the secant convex hull S which denotes the MINKOWSKI set difference of U and V, where U

978-1-5386-8167-1/18$31.00©2018 IEEE

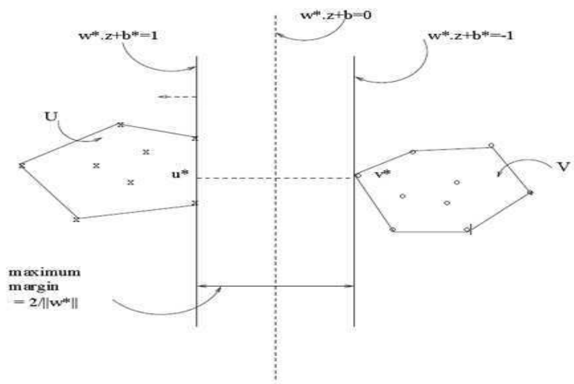


Figure 1: The convex hull of the two classes[10]

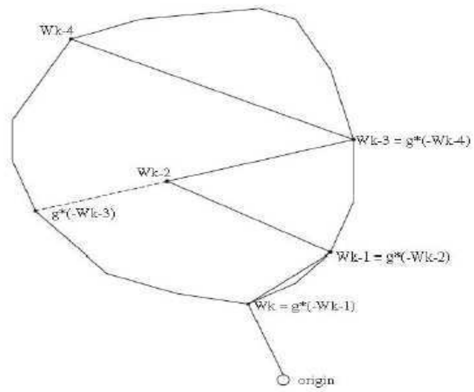


Figure 2: Iteration steps of Gilbert's Algorithm[11]

and V are the convex hulls of each class of training data. S= {s: s = u − v, u $\in$ U, v $\in$ V}: Yu = 1, Yv = -1. The solution to the SVM problem is the point s*, which belongs to the secant convex hull's perimeter and is closest to the origin. Gilbert's Algorithm locates the point of a convex hull closest to the origin with recurring linear steps knowing that s*=u*- v* as shown in figure (1).

The algorithm starts form a random point w_{k-1}, where k is defined as the total number of iterations, then it allocates the point g*(- w_{k-1}), whose projection in the direction of -w_{k-1} is the closest to the origin. This point lies on the secant's perimeter[11]. The goal is to find a point that lies on this segment and can be the closest to the origin. Therefore, there are three cases for this point, it can be the old point w_{k-1}, the new point g*(- w_{k-1}) or a point that lies on the segment between the two mentioned points. In order to identify which case, two parameters called top and bot need to be calculated. Thus, g*(- w_{k-1}) is the point of S that maximizes the inner product with w_{k-1}. This value can be computed by finding g*U and g*V which are the points u and v of classes U and V respectively that maximize the inner products -w_{k-1} . u and w_{k-1}. v : g*(- w_{k-1}) = g*u(-w_{k-1}) − g*v (w_{k-1}). Then it allocates the point w_k which lies on the segment [w_{k-1}, g*(-w_{k-1})] closest to the origin which may not be part of the secant[11].

$$w_k = \begin{cases} w_{k-1} & top \leq 0 \\ g*(-w_{k-1}) & bot \leq top \\ w_{k-1} + \lambda(g*(-w_{k-1}) - w_{k-1}) & otherwise \end{cases} \quad (1)$$

$$top = -w_{k-1} * (g*(-w_{k-1}) - w_{k-1})$$
$$bot = \|g*(-w_{k-1}) - w_{k-1}\|^2$$
$$\lambda = \frac{top}{bot}$$

In the case of top less than bot which defines a point between the old and new point, a circle of radius top and having the origin as the center could be assumed. The excess of bot over the circle can be indicative for lamda value. Thus, from w_{k-1}, a distance lamda on the segment towards g*(- w_{k-1}) defines the point w_k. These steps are repeated till convergence. The terminating condition of Gilbert's is selecting the same point w_k again in a following iteration.

III. Hardware Mapping

In this section, the algorithm in hardware implementation and hardware/FPGA challenges are discussed. The FPGA targeted is Altera Cyclone V.

A. Architecture Design

For SVM training; The training data is stored in separate memories according to their label X = {x_i : y_i = 1}, Y = {x_j : y_j =−1} . Each consists of 3 features, allowing for more parallelism. According to Gilbert's algorithm (mentioned in previous section), it starts with selecting a random point from the two classes. The MINKOWSKI difference of the two data points is calculated and it's the main input to the kernel block along with the training dataset. RBF kernel is deployed, as non-linear SVM is used due to complexity of seizure data points of variety of patients. The output of the kernel is stored in "cache", a memory block module, in which the maximum and minimum points of the two classes are found g*(−w_{k-1}). The norm block operates in parallel to the kernel operation, along with the lamda block calculations. The values of datapoints in cache are dependent on lamda according to the equation (1). The cache data is accumulated for each iteration until the first termination condition is reached, whereby the "CV" module is responsible for storing and searching for any repeated point instances. The $\alpha\beta$ module is for the weighting of each dataset point contribution within the algorithm. The control unit is based on the idea of finite state machine, facilitating transitions between different modes of operation and states. The average of processed data points for each two-consecutive repetition of Gilbert's algorithm is calculated and if the second termination condition is fulfilled, the algorithm terminates; resulting in a legible value for W and b, for classification process.

B. Hardware Challenges

The RBF kernel includes the calculation of exponential, the approach used is LUT. Look Up Tables are elements inside

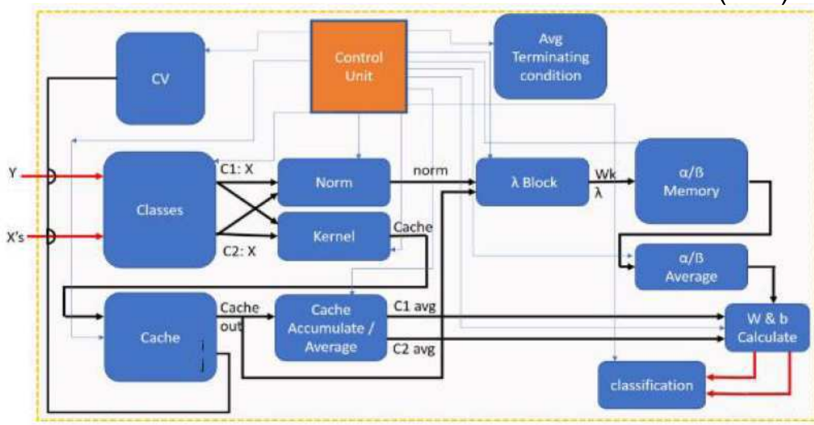


Figure 3: Hardware Block Diagram

the FPGA that would map certain inputs to certain outputs. MATLAB is used to generate a look up table for exp(x). Hence, the "exp" entity was designed such that it represents a ROM loaded with the values of exp(x) where x is interpreted as the address of the memory. In more simple way, this means that going to address "x" in the LUT memory, the value of exp(x) is stored. Another proposed method is Taylor series, but LUT approach is much more effective, resources wise where it proves better performance than the Taylor series approach, however, it over-consumes memory blocks. The targeted FPGA has a total memory of around 5 Mb, which can accommodate this implementation. This block has a latency of only 3 clock cycles. The processing data has wide range of values. Thus, some approximation is required. However, this results in accuracy loss. Fixed point conversion is carried out to maintain sufficient level of accuracy by scaling up numbers. After data observation, the minimum data width is 17 bits [12].

C. FPGA Virtues

The FPGA fabric on-board of the DE10-Nano used is a Cyclone V series which offers a low cost and power consumption. The Development kit supports not only the FPGA but offers the ability to employ HPS (Hard Processor System) in the design as well. The capability of the FPGA to provide a high level of parallelism is employed to solve the training problem in parallel chunks yielding much faster training. In the development phase, assigning the training function to the FPGA off-loads it from the CPU leaving it to process performance measurements simultaneously using the updated parameters that the FPGA provides. This could be done by passing the continuously updated training parameters from the FPGA to the CPU, allowing for hardware acceleration. The FPGA enables high processing of data, as it's needed in Gilbert's algorithm to help the fast convergence of data point, hence, allowing faster response. The concept of implementing Gilbert's algorithm involves dealing with multiple memories and data accumulation, that the FPGA gives a great advantage as it supports fast memory access. The total block memory bits used is 250,320/5,662,720 (4%). Further enhancements in the design were conducted to improve the overall performance; reducing the number of memory slots used, as memory consumes a lot of power. The "CV" module is self-controlled, as it stops the algorithm when it finds a repeated point, as well as, the searching process is done backwards, as there is a much higher probabililty that the repeated point is the last point. The norm calculation was done linearly with inner product instead of using the kernel, as it gives the same performance measures with less complexity of hardware. The design includes plenty of multiplication operations. Multiplication operations often maps to DSP slices when synthesizing. Therefore, and to not fully depend on the target DSP slices number (they are often limited in number) and to give more chances for smaller area, Booth-Multipliers were used. Booth multipliers are known for their fast calculations, the output is calculated in combinational logic (i.e. in the same clock cycle that the inputs change, the output changes accordingly). The division operation also maps to number of DSP blocks. However, it is only possible when the divisor is a constant known number. In some cases, while calculating the average, the divisor is variable. Thus, a division component was implemented with logic units. The total DSP blocks are 12/112 (12%) [11].

IV. Data Set

Result simulations and testing had been conducted on approximate of 24 patients from the CHB-MIT database. Different epochs of seizure period had been used. The performance parameters used for evaluation were as follows:

$$Sensitivity = \frac{TP}{TP+FN}$$
$$Specificity = \frac{TN}{TN+FP}$$
$$Accuracy = \frac{TP+TN}{TP+FN+FP+TN}$$

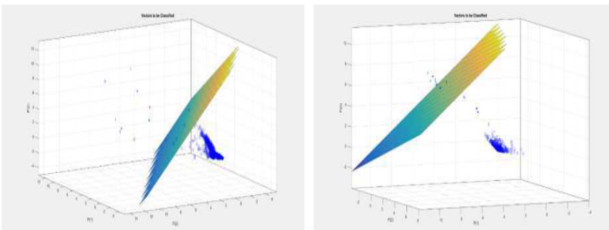
The simulations have shown an average values for accuracy of 90.96116 % , specificity of 90.95732 % and sensitivity of 74.35 %.

V. Results and Discussion

As mentioned in Table 1, primary MATLAB Algorithm simulation results are discussed in this context for different patients in Figure (4) compared to the RTL simulation of the same patient cases. From observing the previous figures, patient 1 RTL simulations gave promising results that were close to the results obtained from MATLAB simulations since feature space of patient 1 is less overlapping. Although some approximations were carried out in simulating RTL, the algorithm could find the proper support vectors to draw the hyperplane in between. On the other hand, seizure and non-seizure points of patient 5 are more overlapping in the feature

Table I: Seizure Detection Results of Different Algorithms

Patient	Turky N. Alotaiby[13]			Gilbert's Algorithm							
	CSP			Hardware				Software			
	Sensitivity	Specificity	Accuracy	Sensitivity	Specificity	Accuracy	Time	Sensitivity	Specificity	Accuracy	Time
1	100	90	95.238	87.85046	84.21146	84.22298	167.162μs	91.5888	77.4763	77.521	3.1406 s
3	-	-	-	88.63636	88.6605	88.6605	170.77μs	90.9091	84.884	84.8927	7.9062 s
5	100	100	100	46.5632	100	100	187.01μs	90.4348	88.9032	88.9091	3.1406 s



(a) Patient 1 (b) Patient 5

Figure 4: Hyperplane of patients 1 and 5 showing both seizure and non-seizure points.

space which caused more complications in finding the proper margin in the presence of the approximations. Hence, gave lower results in comparison with MATLAB simulation. This is obvious from the previous hyperplane figure of patient 5 in MATLAB which tends to be favoring non-seizure points. This shift to the upside towards distant seizure points introduced severe error which reflected on the performance parameters. The C tilde parameter in our analysis can always be adjusted for each patient to get better performance results. C tilde is an input that can be changed easily before training and classifying the data for each patient. Some patients obtained good results with C tilde that is equal to 0.0016 and others with C tilde that is equal to 0.001.

The logic utilization is equal $\frac{9144}{41910}$which is 22% of the FPGA ALMs. The power analysis is conducted after successful and high confidence estimation using sufficient toggle rates provided by testbenches. The calculated core dynamic power dissipated was 90.33 mW.

Also, the acceleration factor can be calculated from the training time as mentioned in Table 1.

VI. Conclusion

The problem of seizure detection using support vector machine (SVM) was addressed to classify between seizure and non-seizure classes. The SVM goal is to be able to draw an accurate hyperplane between the two classes to categorize the signals of a patient's brain to be correctly detected. The main focus of the work was proving that Gilbert's Algorithm - the geometrical approach used long time ago in few applications - is able to be a good classifier for seizure detection problems. Burning Gilbert's training algorithm and classifier on FPGA is done successfully. This is very much the key component in future attempts to build the chip using ASIC technology. Future work concerns deeper analysis in the ASIC field in order to build the chip that would contribute in seizure prevention. Many optimizations have been left for the future work to obtain the optimal minimum power. The target of the chip is to consume a low power that the battery can survive long periods. A future vision in the design problem is to store all the candidate training algorithms in the chip and use partial dynamic reconfiguration to let the chip use the most efficient algorithm. (i.e. Low power required -> reconfigure to Algorithm(1), Fast Training required -> reconfigure to Algorithm(2)).

Acknowledgment

This work was partially funded by ONE Lab at Zewail City of Science and Technology, Egypt and Cairo University, Egypt and Information Technology Academia Collaboration (ITAC), Egypt.

References

[1] M. England, A. Schultz, C. Liverman, and L. Strawbridge, "Epilepsy across the spectrum: Promoting health and understanding: A summary of the institute of medicine report," vol. 12, no. 1, pp. 108–115, 2012.

[2] A. Varsavsky, I. Mareels, and M. Cook., *Epileptic Seizures and the EEG*, 2011.

[3] M. A. Elgammal, O. A. Elkhouly, H. Elhosary, M. E. Sayed, A. Mohieldin, K. N. Salama, and H. Mostafa, "Linear and nonlinear feature extraction for neural seizure detection," no. 1, pp. 795–798, 2018.

[4] K. Helal, A. Y. Aboelmakarem, A. M. B. R. Fouad, T. S. Kamel, K. A. M. Nageeb, M. M. K. Mohamed, M. M. Abdelrhman, Y. Ismail, and H. Mostafa, "Low-power high-accuracy seizure detection algorithms for neural implantable platforms," no. 1, pp. 231–234, 2017.

[5] G. Maximous, A. El-Gunidy, H. Mostafa, T. Ismail, and S. Gabran, "A new sensitivity-specificity product-based automatic seizure detection algorithm," no. 1, pp. 115–118, 2017.

[6] R. Ramirez-Padron, "A roadmap to svm sequential minimal optimization for classification," 2007.

[7] S. S., "Gradient descent: All you need to know," 2018.

[8] N. Donges, "Gradient descent in a nutshell," 2018.

[9] S. Martin, "Training support vector machines using gilbert algorithm," 2005.

[10] S. S. Keerthi, S. K. Shevade, C. Bhattacharyya, and K. R. Murthy, "A fast iterative nearest point algorithm for support vector machine classifier design," *IEEE transactions on neural networks*, vol. 11, no. 1, pp. 124–136, 2000.

[11] M. Papadonikolakis and C.-S. Bouganis, "Efficient fpga mapping of gilbert algorithm for svm training on large-scale classification problems," 2008.

[12] A. Taylor, "The basics of fpga mathematics," 2012.

[13] T. N. Alotaiby, S. A. Alshebeili, F. E. A. El-Samie, K. H. Aljibreen, and E. Alkhane, "Seizure detection with common spatial pattern and support vector machines," 2015.