

# Low-Power Analog Integrated Architecture of the Voting Classification Algorithm for Diabetes Disease Prediction

Vassilis Alimisis *Student Member, IEEE*, Charis Aletraris *Student Member, IEEE*, Nikolaos P. Eleftheriou *Student Member, IEEE*, Emmanouil Anastasios Serlis *Student Member, IEEE*, Alex James *Senior Member, IEEE*, Paul P. Sotiriadis *Fellow, IEEE*

Department of Electrical and Computer Engineering, National Technical University of Athens, 15780 Athens, Greece

School of Electronic Systems and Automation, Digital University Kerala, Trivandrum 695317, India

**Abstract**—A low-power ( $\sim 600\text{nW}$ ), fully analog integrated architecture for a voting classification algorithm is introduced. It can effectively handle multiple-input features, maintaining exceptional levels of accuracy and with very low power consumption. The proposed architecture is based on a versatile Voting algorithm that selectively incorporates one of three key classification models: Bayes or Centroid, or, the Learning Vector Quantization model; all of which are implemented using Gaussian-likelihood and Euclidean distance function circuits, as well as a current comparison circuit. To evaluate the proposed architecture, a comprehensive comparison with popular analog classifiers is performed, using real-life diabetes dataset. All model architectures were trained using Python and compared with the software-based classifiers. The circuit implementations were performed using the TSMC 90 nm CMOS process technology and the Cadence IC Suite was utilized for the design, schematic and post-layout simulations. The proposed classifiers achieved sensitivity of  $\geq 96.7\%$  and specificity of  $\geq 89.7\%$ .

**Index Terms**—Activation Function Circuit, Analog VLSI, Biomedical circuits, Low power design, Machine Learning circuits.

## I. INTRODUCTION

THE integration of Machine Learning (ML) and Artificial Intelligence (AI) in bioengineering is an essential catalyst for reshaping research, diagnostics, and treatment approaches [1]. The complexity of biological systems requires innovative methods to uncover intricate patterns and predict their outcomes. ML and AI techniques are used to navigate complex datasets, revealing insights that conventional methods may miss [2]. These technologies assist in disease detection, genomic analysis, protein folding predictions, drug discovery, and personalized medicine, providing bioengineers with predictive models that streamline experimentation and reduce trial and error [2], [3]. Moreover, the impact of ML and AI lies in their ability to translate theoretical insights into practical applications, thereby enhancing precision and effectiveness. Furthermore, emerging techniques focus on the transformation brought about by ML and AI in the field of bioengineering, accentuating their role in reshaping the comprehension and manipulation of biological systems to enhance health [4].

Alongside the advancement of ML and AI in biomedical applications, the importance of cutting-edge hardware solutions

has surfaced as an essential synergy [5]. The intricate demands for processing extensive datasets, intricate simulations, and real-time analyses inherent in bioengineering call for hardware architectures that seamlessly complement the cognitive capabilities of ML and AI [6]. High-performance computing systems, specialized hardware accelerators, and advanced sensor technologies provide the computational power and precision required to manage the complexities of biological phenomena [6]. These hardware strides not only expedite the execution of ML and AI algorithms but also pave the way for novel methodologies that smoothly integrate data acquisition, processing, and feedback mechanisms [7]. Covering genomics, proteomics, and more, the fusion of ML, AI, and purpose-built hardware empowers researchers and practitioners to delve further into the intricacies of biological systems [8]. This synergy lays the foundation for pioneering discoveries and innovations that significantly impact fields such as medicine, agriculture, and environmental sustainability.

Analog computing techniques have recently gained renewed attention in biomedical applications, serving as an innovative approach to augment ML methodologies [9]. Presenting a promising avenue for addressing the computational demands of complex ML tasks in biomedicine, analog computing demonstrates the ability to process continuous signals in real-time and leverage the inherent parallelism of physical systems [10]. This is particularly relevant in scenarios in which precision, energy efficiency, and low latency are crucial factors. The suitability of analog computing for processing biological signals and mimicking physiological processes aligns well with the intricacies of medical data analyses [11]. By leveraging the capacity of analog computing to directly manipulate and process continuous signals, researchers can potentially achieve faster and more energy-efficient ML inference for applications such as real-time diagnostics, wearable health monitoring, and neurocomputing [12], [13]. The fusion of analog computing with ML in biomedicine holds promise for unlocking novel insights, facilitating faster decision-making, and enhancing the overall efficiency of data-driven medical interventions [14].

In the literature there are a variety of analog hardware classifiers including, cascaded-connected Bayes [15], Gaussian mixture model (GMM) [16], Radial Basis Function (RBF)

[17], RBF-Neural Network (NN) [18], Support Vector Machine (SVM) [19], [20], Multilayer Perceptron (MLP) [21], K-means [22], a Support Vector Regression (SVR) [23], Self-Organized Map (SOM) [24], Long Short-Term Memory (LSTM) [25], a Fuzzy [26], Threshold [27] and cascaded-connected Centroid classifier [28]. For more information on analog and mixed signal classifiers, please refer to [29], where these classifiers are summarized and explained. Compared to this work, related studies [15]–[17], [19], [20], [24], [26]–[28] lack the ability to control weights for each separate feature; instead, they can only adjust the overall probability for the entire class. The operating range of classifiers in the existing methodologies is limited. However, with the voting classification algorithm, there is greater flexibility because of the absence of a cascaded format connection (regarding circuits). This offers the following advantages. This method can handle a large number of features eliminating the need for Principal Component Analysis (PCA) [30]. Furthermore, it offers the potential for each circuit to operate with minimal consumption (by utilizing minimum biasing values, e.g.  $I_{bias}$ ). As a result, it exhibits significantly lower power consumption than existing methods, especially when dealing with a larger number of features. All of these aspects are discussed in Section VI.

Motivated by the low-power and area efficiency requirements of biomedical smart sensor systems [31], [32], numerous analog feature extraction architectures [33]–[36] and the different approaches regarding voting classifiers [37]–[40], this study proposes an alternative, low-power, and analog integrated architecture based on a Voting classification algorithm. Demonstrating considerable promise as a classifier suitable for battery-dependent biomedical smart sensor classification systems, the implemented design attains high accuracy. This design was meticulously crafted and validated using real-world diabetes disease prediction dataset [41]. Post-layout simulations conducted in a TSMC 90nm CMOS process via Cadence IC Suite validated the accuracy of the devised implementation, compared to a software-based counterpart. Moreover, to ensure comprehensiveness, this study incorporates an exhaustive comparative analysis of the proposed classifier against analog classifiers.

The remainder of this study is structured as follows. In Section II, we delve into the essential mathematical background of the Voting Model used for classification. Section III presents an analysis of both the high-level architecture of the proposed classifier and the transistor-level implementations of the fundamental building blocks. The training and tuning capabilities of the proposed architecture are described in Section IV. In Section V, we assess the accuracy of the classifier using a real-world diabetes dataset. Section VI provides a comparison study with related analog classifiers and summarizes and discusses the main aspects. Finally, Section VII provides concluding remarks.

## II. BACKGROUND

In this section, the mathematical background of both the Voting algorithm and the corresponding ML models (Bayes,

Centroid, and Learning Vector Quantization) which are the theoretical foundations of the proposed building blocks are analyzed. Each ML model is associated with a distinct activation function, implemented in the hardware.

The Voting algorithm was implemented in this study, because of its simplicity and interpretability [42]. The two types of voting algorithms are described as follows; Hard Voting and Soft Voting [42], [43]. The Hard Voting algorithm, similar to the conventional approach, involves combining class labels based on the majority vote. However, it acknowledges the binary nature of a decision by considering only the final outcome. It does not account for the confidence levels of individual classifiers or the probabilistic nature of their predictions. In contrast, the Soft Voting algorithm considers the probabilities or confidence levels assigned by each classifier for different classes. Rather than relying solely on majority voting, it combines the probabilities across all classifiers and calculates a weighted average. This approach enables a more nuanced decision-making process by considering the confidence levels of classifiers. By incorporating probabilistic information, Soft Voting algorithms tend to be more robust and accurate in handling classification tasks.

For the analog classifier described here, a Soft-Voting algorithm was used [42]. The classifier consists of multiple voters ( $V_i$ ), where  $i \in \{1, \dots, N_f\}$  and  $N_f$  represents the number of voters corresponding to classes  $C_{k_i}$ ,  $k \in \{1, \dots, N_{cla}\}$  and  $N_{cla}$  represents the number of classes. Assuming Gaussian distribution (we also use the Mahalanobis distance), the voting strength of an input vector  $V$  with respect to the class  $C_k$ ,  $k \in \{1, \dots, N_{cla}\}$  is interpreted as:

$$\mathcal{S}_{C_k}(V) = \sum_{i=1}^{N_f} \frac{1}{\sqrt{2\pi\sigma_{k_i}^2}} e^{-\frac{(V_i - \mu_{k_i})^2}{2\sigma_{k_i}^2}} \quad (1)$$

where  $\mu$  is the mean, and  $\sigma$  is the standard deviation.

Subsequently, the votes are linearly combined, allowing a voter to allocate the percentages of their vote across different classes, as described in:

$$\mathbb{F}_{C_k}(V) = \frac{\mathcal{S}_{C_k}(V)}{\sum_{k=1}^{N_{cla}} \mathcal{S}_{C_k}(V)} \quad (2)$$

For instance, a voter may assign 40%, 50% and 10% to the first, second and third classes respectively. This process is carried out for all voters and the class with the majority of votes is selected as the winning class as interpreted by the arg max operator:

$$\theta = \arg \max_k \mathbb{F}_{C_k}(V) \quad (3)$$

where  $\theta$  represents the class with the highest cumulative vote.

### A. Bayes Model

In recent years, there has been a growing interest in the Bayes model across various fields [44], [45]. This approach provides a probabilistic framework that integrates prior knowledge with uncertainty estimation. By employing Bayes' theorem, it updates classification probabilities using observed data. The model considers both the prior estimation of the data

under classification and the probability of observing specific features or patterns in the input data. One notable advantage of the Bayes model is its effective handling of uncertainty using probability distributions.

The Bayes probabilistic model [46], [47] is described by:

$$\mathbb{P}(C_k|X) = \frac{\mathbb{P}(C_k) \cdot \mathbb{P}(X|C_k)}{\mathbb{P}(X)} \quad (4)$$

where  $C_k$  denotes the  $k$ -class ( $k \in \{1, \dots, N_c\}$ ),  $N_c$  represents the number of classes and  $X$  is the input vector.

- $\mathbb{P}(C_k|X)$  is the posterior (*a-posteriori*) distribution of  $C_k$  class given the observations in  $X$ .
- $\mathbb{P}(C_k)$  is the prior (*a-priori*) distribution of  $C_k$  class.
- $\mathbb{P}(X|C_k)$  is the likelihood of observing  $X$  given the distribution of  $C_k$  class.
- $\mathbb{P}(X)$  is the probability of observing  $X$  (evidence probability), which is also called the marginal likelihood or the normalization constant. This ensures that the posterior distribution is integrated into 1.

The likelihood function for an independent Gaussian distribution is the product of the probability density function evaluated at each observation  $C_{k_n}$ .

$$\mathbb{P}(X|C_k) = \prod_{n=1}^{N_f} f(\mathbf{X}_n|C_{k_n}) \quad (5)$$

where  $N_f$  denotes the number of features in each class. For a Gaussian distribution the probability density function is

$$f(x|\mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (6)$$

Combining the above equations, we obtain

$$\mathbb{P}(X|C_k) = \prod_{n=1}^{N_f} \frac{1}{\sqrt{2\pi\sigma_{k_n}^2}} e^{-\frac{(X_n - \mu_{k_n})^2}{2\sigma_{k_n}^2}} \quad (7)$$

Applying the Maximum Likelihood Estimation (MLE) method results in the hypothesis that has the greatest probability of being true. To this end, the values of the parameters that maximize the posterior probability of  $C_k$  class are computed as follows:

$$\theta = \arg \max_k \mathbb{P}(C_k|X) = \arg \max_k \mathbb{P}(C_k) \cdot \mathbb{P}(X|C_k) \quad (8)$$

where  $\theta$  represents the MLE of the parameters of  $C_k$ . The evidence probability  $\mathbb{P}(X)$  does not affect the result, because it serves only as a normalization constant. The main reason is that in each class the evidence probability is the same.

### B. Centroid Model

The Centroid Classifier (CC) is an essential example of a centroid-based classification model [48]. In this paradigm, each class  $i$  is uniquely represented by vector-form centroid ( $\mathbf{c}_i$ ). Consequently, an input vector, also represented as  $\mathbf{q}$ , is assigned label  $y(\mathbf{q})$  of the class whose centroid bears the closest proximity to it, as documented in [49], [50]. This process was captured succinctly using the following equation:

$$y(\mathbf{q}) = \operatorname{argmin}_{i \in \{1, 2, \dots, N\}} \|\mathbf{q} - \mathbf{c}_i\|_2, \quad (9)$$

where  $N$  denotes the number of classes. Another alternative is to evaluate the similarity between the input  $\mathbf{q}$  and each centroid  $\mathbf{c}_i$ , which then informs the label  $y(\mathbf{q})$ :

$$y(\mathbf{q}) = \operatorname{argmax}_{i \in \{1, 2, \dots, N\}} \operatorname{sim}(\mathbf{q}, \mathbf{c}_i), \quad (10)$$

Here,  $\operatorname{sim}()$  is a chosen similarity function, for example the Mahalanobis distance [51].

Expanding upon the CC, the Multiple Centroid Classifier MCC accommodates the assignment of multiple centroids (clusters) to each class. This enhancement propels the classifier's accuracy. Consequently, an input vector  $\mathbf{q}$  is allocated a label  $y(\mathbf{q})$  as per the equation:

$$y(\mathbf{q}) = \operatorname{argmax}_{\substack{i \in \{1, 2, \dots, N\} \\ j \in \{1, 2, \dots, K_i\}}} \operatorname{sim}(\mathbf{q}, \mathbf{c}_{ij}), \quad (11)$$

where  $K_i$  represents the number of centroids assigned to the class  $i$ . This extended methodology demonstrates the adaptability of the classifier in handling diverse and intricate classification scenarios.

### C. Learning Vector Quantization Model

Learning Vector Quantization (LVQ) is a prototype-based classification algorithm [52]. In LVQ, a set of weight vectors are employed as prototypes, and these prototypes are defined within the feature space of the observed data. An LVQ can be conceptualized as a two-layer neural network architecture, where the weight vectors in the first layer represent these prototypes. This approach allows LVQ to make classification decisions based on the similarity between input data points and prototype vectors, making it a valuable tool in pattern recognition and machine learning tasks. Consequently, LVQ facilitates a user-friendly representation of the input data, which is especially beneficial for experts within the respective application domain. Additionally, it can be readily extended to handle multi-class problems [53]. LVQ is widely used in ML, finding applications in diverse areas, ranging from Magnetic Resonance Imaging (MRI) segmentation [54], detection of seizure activity in EEG [55] and prediction of laser butts [56] to COVID-19 diagnosis [57]. These applications underscore their significance, making hardware development and implementation a worthwhile endeavor.

The LVQ model [52] is initiated by assigning a set of  $N_d$  codebook vectors to each of the  $N_{cla}$  distinct classes. It then proceeds to identify the codebook vector  $m_i$  with the smallest Euclidean distance (for this work Mahalanobis distance) to the input sample  $x$ . Subsequently, sample  $x$  is categorized as belonging to the class  $N_{cla}$  associated with its nearest codebook vector  $m_i$ . This classification was performed using the following equation:

$$c = \operatorname{argmin}_{i \in \{1, 2, \dots, N_v\}} \|m_i - x\|, \quad (12)$$

where  $\|\cdot\|$  denotes the Euclidean norm. The information pertaining to the winning class  $c$  serves a dual purpose during both inference and training. It plays a crucial role in correctly classifying the input data  $x$ . Furthermore, it guides the

updating of codebook vectors (feature arrays) during training based on whether  $x$  and  $m_i$  belong to the same class. The updating process is described by the following equation:

$$m_c(t+1) = \begin{cases} m_c(t) + a(t)(x(t) - m_c(t)), & c_m = c_x \\ m_c(t) - a(t)(x(t) - m_c(t)), & c_m \neq c_x \end{cases} \quad (13)$$

Here,  $a(t) \in [0, 1]$  represents the learning rate.

### III. PROPOSED ARCHITECTURE

In this section, we analyze the proposed analog high-level architecture of the voting classifier, along with the basic building blocks. Specifically, the proposed architecture can implement several ML models. We selected Bayes, Centroid and Learning Vector Quantization (LVQ) classifiers to demonstrate the voting architecture. To this end, we introduce an Activation Function Circuit (AFC) that can selectively realize either the Gaussian function or Mahalanobis distance function depending on the particular ML model.

The proposed voting architecture is versatile, accommodating various numbers of classes (which corresponding to prototypes for LVQ) and input dimensions that represents the number of features. To delve into the architecture of the proposed classifier, let's consider a scenario involving a problem with  $N_{cla}$  classes and  $N_d$  input dimensions. An example of the  $k$ -th class,  $k \in \{1, 2, \dots, N_{cla}\}$  of the proposed classifier is shown in Fig. 1. It comprises two basic blocks; an AFC and a cascode Current Mirror (CM). Each class consists of  $N_d$  AFCs, each of which describes an activation function (Gaussian or Mahalanobis distance) in the domain space of the classification problem, as explained thoroughly in Section II. In addition, using the Soft-Voting model, described in Eq. (1), a summation of the output currents, representing the probability density function, is required. This summation of the classes' output nodes is performed through the cascode CMs. These were utilized to minimize potential distortions in the calculations that might arise from undesirable effects on the output currents of the AFCs.

The analog integrated architecture of the proposed voting classifier is shown in Fig. 2, where  $I_{in}$  and  $I_r^{(k)}$ ,  $k \in \{1, 2, \dots, N_{cla}\}$  represent the input and parameter vectors respectively, that is  $I_{in} = \{I_{in,j}\}_{j=1}^{N_d}$  and  $I_r^{(k)} = \{I_{r,j}^{(k)}\}_{j=1}^{N_d}$ . It consists of  $N_{cla}$  class cells, as described above and their outputs are fed into a Winner-Take-All (WTA) circuit to determine the winning class.

Because a low-power design is one of the main goals of this work, all transistors operate in the sub-threshold region, with the power supply rails set to  $V_{DD} = -V_{SS} = 0.3V$ . The selection of the basic building blocks and power supply rails is guided by a trade-off between achieving high accuracy, minimizing power consumption, and ensuring the correct operating principles for the entire classifier. In addition, we ran noise-transient simulations to verify the behavior of the proposed classifier. The classification results appear to be robust. To a certain extent this indicates us that the errors owing to internal noise are small concerning the errors in the data. Moreover, the ease of implementation of the

Gaussian function and Mahalanobis distance circuits, renders them favorable candidates for area efficient and low-power classifiers.

The selection of dimensions is a multiparametric process, as reducing  $W$  and  $L$  leads to a decrease in the overall area. Moreover, increasing  $L$  improves the system's noise, reduces leakage current, and decreases the overall bandwidth (BW). On the other hand, increasing  $W$  enhances noise, increases leakage current, and reduces BW. Additionally, increasing the dimensions reduces mismatch between devices. Based on the above considerations and aiming for correct biasing, the selection of dimensions in the transistors was made.

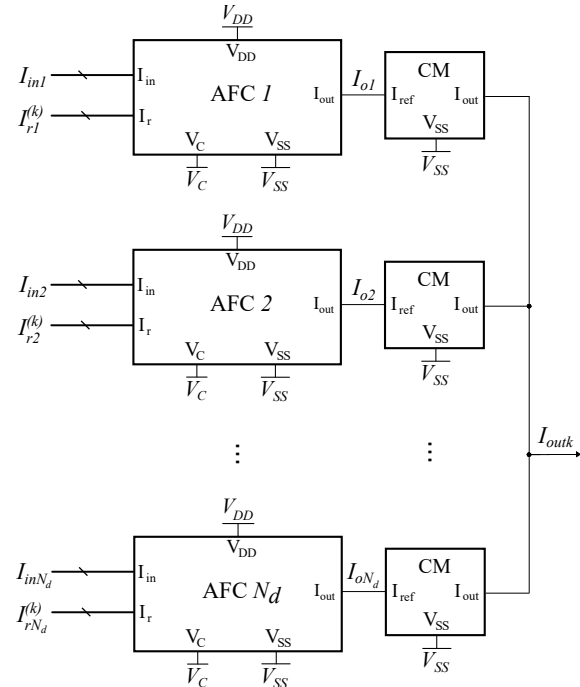


Fig. 1: The implementation of the  $k$ -th class,  $k \in \{1, 2, \dots, N_{cla}\}$ , of the proposed classifier. It comprises  $N_d$  AFCs that describe the similarity functions for each feature, along with an equal number of cascode CMs used to implement the current summation on the output.

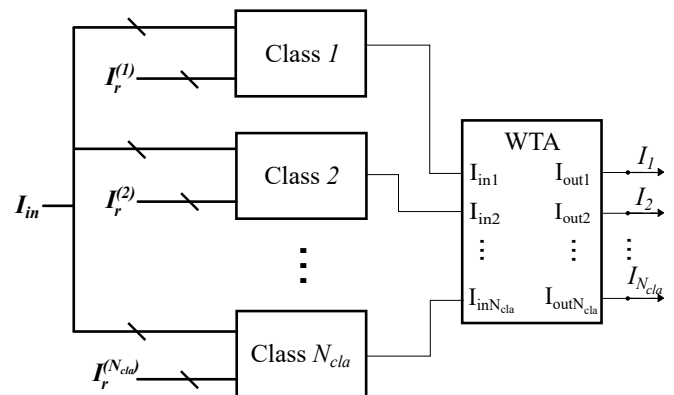


Fig. 2: The proposed classifier architecture consisting of  $N_{cla}$  class cells, as depicted in Fig. 1, and a WTA circuit that determines the winning class.

### A. Current-mode Gaussian function circuit

To implement a Gaussian function as an activation function a current-mode Gaussian function circuit [15] was employed as the AFC, as depicted in Fig. 3. The characteristics of the Gaussian function that is generated, such as the mean value, variance, and height, can be manipulated through electronic control by utilizing the parameters of the implemented Gaussian function circuit. Specifically, the height of the Gaussian function was determined by the bias current  $I_{bias}$ , as shown in Fig. 4(left). The mean value is regulated by the current  $I_r$ , which influences the modified differential pair as shown in Fig. 4(right). As for the variance, there exist two methods to manage it; alteration of the bulk voltage  $V_c$  for transistors  $M_{nd1}$  and  $M_{nd2}$  (deep-n-well transistors in a triple-n-well technology) in the differential pair, as shown in Fig. 5(left) and manipulation of the size ratio  $M$  (number of multipliers in  $M_{n5} - M_{n8}$ ) between the neuron transistors in the WTA circuit, as depicted in Fig. 5(right). By employing deep-n-well transistors, we achieve higher accuracy by setting a specific variance for each AFC based on the training data. This was achieved through the trade-off of a slight increase in area. The dimensions of the transistors in the current-mode Gaussian function circuit are listed in Table I.

TABLE I: Transistors' Dimensions for the Gaussian function circuit (Fig. 3).

Transistors	W/L ( $\mu m/\mu m$ )	Transistors	W/L ( $\mu m/\mu m$ )
$M_{n1}-M_{n3}$	0.4/1.6	$M_{n4}$	0.8/1.6
$M_{n5}, M_{n7}$	1.2/1.6	$M_{n6}, M_{n8}$	1.2/3.2
$M_{nd1}, M_{nd2}$	0.4/3.2	$M_{p1}-M_{p6}$	0.4/1.6

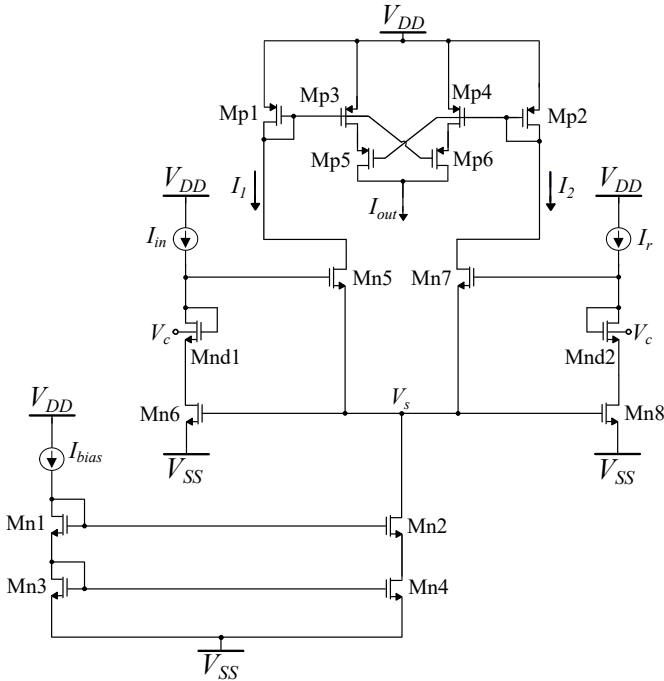


Fig. 3: Gaussian Function circuit as AFC. The currents  $I_{in}$ ,  $I_r$  and the voltage  $V_c$  correspond to the classifier's input signal, mean value and variance respectively. Also the bias current  $I_{bias}$  control the height of the Gaussian function.

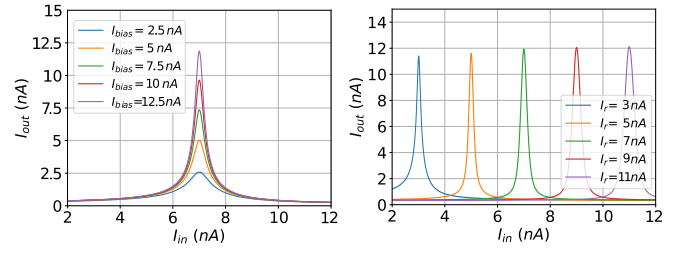


Fig. 4: The output current of the Gaussian function circuit as a function of  $I_{in}$  and parameterized on  $I_{bias}$ , for  $I_r = 7nA$ ,  $V_c = 180mV$  and  $M = 1$  (Left). The output current of the Gaussian function circuit as a function of  $I_{in}$  and parameterized on  $I_r$ , for  $I_{bias} = 12nA$ ,  $V_c = 220mV$  and  $M = 1$  (Right).

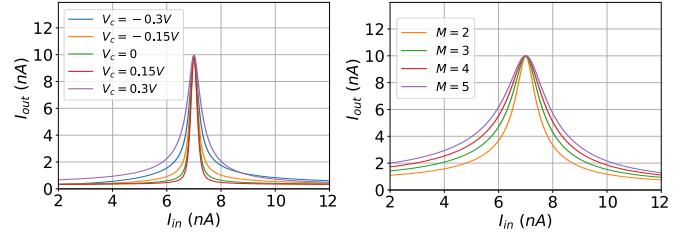


Fig. 5: The output current of the Gaussian function circuit as a function of  $I_{in}$  and parameterized on  $V_c$ , for  $I_{bias} = 10nA$ ,  $I_r = 7nA$  and  $M = 1$  (Left). The output current of the Gaussian function circuit as a function of  $I_{in}$  and parameterized on  $M$  (number of multipliers), for  $I_{bias} = 10nA$ ,  $I_r = 7nA$  and  $V_c = 180mV$  (Right).

### B. Mahalanobis distance circuit

To realize the Euclidean distance function as a similarity function, relying on Eq.(10), a specialized circuit was employed as AFC. This circuit was designed to calculate the Mahalanobis distance in a current-mode fashion, as illustrated in Fig. 6. This particular circuit operates using the translinear principle [58], which is a technique commonly used in analog circuits for various computations. The mathematical expression that underlies the Mahalanobis distance circuit is based on the ratio of the two currents  $I_{in}$  and  $I_r$ :

$$I_{out} = \frac{I_{in}^2}{I_r} \quad (14)$$

The output current of the Mahalanobis distance circuit is tuned via the current parameter  $I_r$  and the voltage parameter  $V_c$ , as illustrated in Fig. 7. When it comes to performing summation in the current domain, the Mahalanobis distance circuit offers a straightforward approach. This is accomplished by physically connecting wires that carry the currents to be summed. This direct current-based summation mechanism is a characteristic of current-mode circuits and simplifies the overall computation process. However, to improve the quality of the aforementioned summation, cascode current mirrors were employed, as mentioned earlier in this Section. The dimensions of the Mahalanobis distance circuit transistors were set to  $\frac{W}{L} = \frac{3.2\mu m}{1.6\mu m}$ .

### C. Cascaded Winner-Take-All Circuit

The process of identifying the class with the majority of votes involves utilizing a distance comparison circuit, specif-

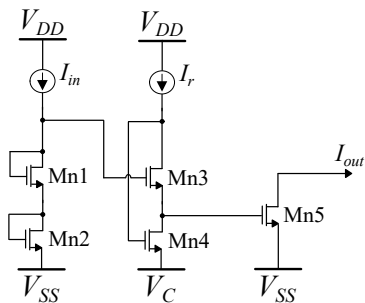


Fig. 6: Translinear circuit for computing  $\frac{I_{in}^2}{I_r}$  needed to obtain the Mahalanobis distance. It is used as an AFC.

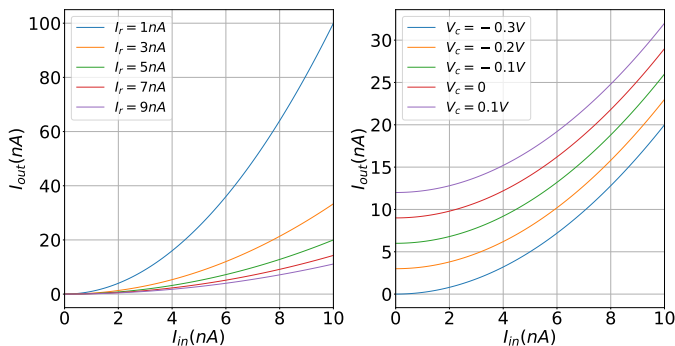


Fig. 7: The output current of the Mahalanobis distance circuit as a function of  $I_{in}$  and parameterized on  $I_r$  for  $V_c = V_{SS} = -300mV$  (Left). The output current of the Mahalanobis distance circuit as a function of  $I_{in}$  and parameterized on  $V_c$  for  $I_r = 3nA$  (Right).

ically referred to as a WTA circuit (argmax operator circuit) [59]. In the context of a classification task involving  $N_{cla}$  classes, the conventional Lazzaro WTA circuit comprises of  $N_{cla}$  neurons. As shown in Fig. 8, these neurons share a common bias current. Each neuron in the WTA circuit corresponds to a distinct class. The functionality of the WTA circuit revolves around effectively recognizing the class associated with the highest input current, subsequently assigning a non-zero output current to the corresponding neuron, while the remaining neurons provide an output current of zero.

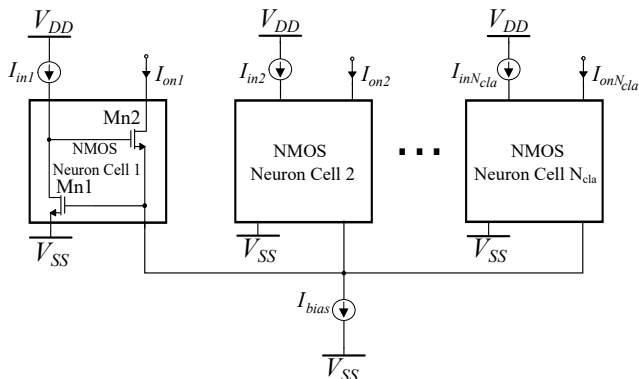


Fig. 8: A  $N_{cla}$ -neuron Standard Lazzaro NMOS Winner-Take-All (WTA) circuit.

In situations where multiple input currents exhibit comparable magnitudes, the circuit functions within its linear

region, potentially leading to the emergence of multiple winners. However, this outcome is generally undesirable in most classification scenarios. To address this concern, an enhanced WTA circuit is used, as shown in Fig. 9. Notably, this modified design involves a cascaded arrangement of three WTA circuits, following a structure similar to that presented in [16]. By alternating the employment of the NMOS and PMOS designs, the need for interconnecting elements (i.e. current mirrors) between consecutive WTA circuits is obviated. The dimensions of the transistors in the Cascaded WTA circuit were set to  $\frac{W}{L} = \frac{0.4\mu m}{1.6\mu m}$ .

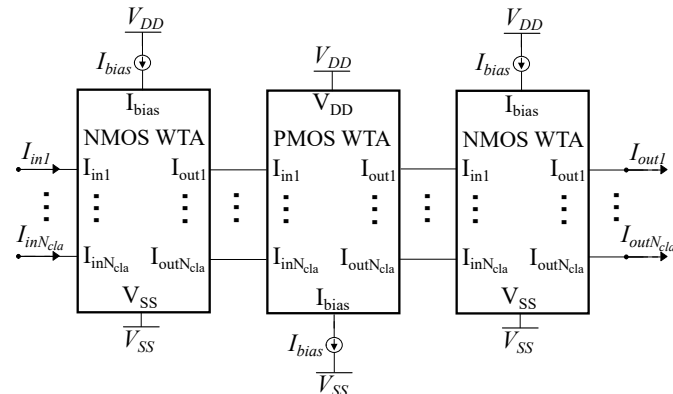


Fig. 9: The implemented Cascaded WTA circuit built by alternating the simple PMOS and NMOS WTA designs. It provides more accurate results because it can deal with ambiguous cases.

#### IV. TRAINING AND TUNING CAPABILITIES

The previously mentioned voting hardware architecture was developed based on AFC, serving as a distance metric for the prototypes within each class. This approach allows for the utilization of electronically tunable parameters, namely  $I_r$  and  $V_c$ , which can be employed to create a post-layout classification chip. This chip offers easy adjustability, making it adaptable to specific requirements of each application. Furthermore, the tunability of the system was expanded to such an extent that the designed classifier became versatile. This versatility enables the classifier to effectively address a wide array of classification challenges, regardless of the input dimensions ( $N_d$ ) or quantity of classes ( $N_{cla}$ ).

##### A. Offline training

Initially, the voting procedure utilized software to determine the circuit parameters' values. Unfortunately, this type of circuit has inherent limitations. To address this issue, a linear approximation is introduced to establish the range for each feature within the necessary framework. The dataset file adopts an AFC operational range of [3, 9] nA, achieving an optimal balance between accuracy, minimal power consumption, and space utilization. During the approximation process, a delicate trade-off occurs between information preservation and range proportionality. This arises from the AFC's capability to attain an optimal output solely for a particular  $I_{bias}$ , with any departure from this bias current leading to a reduction in accuracy. Moreover, augmenting the supplier values helps alleviate

the accuracy decline stemming from  $I_{bias}$  deviation, yet it concurrently enlarges the area and power supply demands.

It's important to recognize that selecting a lower range presents a challenge, as it narrows down the voting range for each AFC excessively. This restricted scope complicates the accurate detection of subtle differences. On a positive note, a notable advantage of this algorithm is its ability to leverage the full allowable range of AFC. This adaptability is primarily enabled by the autonomy of each AFC, which is achieved through the utilization of a cascode current mirror after them.

The chosen dataset is accessible in digital format, with its features pre-processed to accommodate the circuit's operational range. Following this, the software-based classifier undergoes training on the dataset for a specific number of epochs to attain the highest accuracy and desirable cluster/centroids. The selection of the optimal number (preferring the minimum possible) of clusters/centroids is conducted in software, since it is a hyperparameter of the classifier, targeting the best accuracy [60] (avoiding overfitting).

In hardware tuning, the objective is to reduce power consumption while maintaining high classification accuracy levels. For our classification task, the number of centroids/clusters was set to  $N_{clu} = 1$  per class, as this choice results in lower consumption. However, determining the values of  $V_c$  and  $I_{bias}$  presents a wide range of possibilities. A lookup table is employed to establish the relationship between the height or variance (minimum value) and their respective circuit parameters,  $I_{bias}$  or  $V_c$ . The Gaussian function circuit facilitates the customization of both the height ( $I_{bias}$ ) and variance ( $V_c$ ) of the Gaussian curve, utilizing the training data. In contrast to Mahalanobis, where only  $V_c$  is adjusted, this modification influences the curve's minimum. To ensure a fair comparison among the three machine learning models, we chose to set constant values for  $V_c$  and  $I_{bias}$  based on power consumption, rather than utilizing the training data, which would be the optimal approach. This decision was made due to the trade-off of sacrificing a slightly lower accuracy for the implemented models.

Since this simplified voting algorithm, which ensures a fair comparison among implemented models, lacks a direct method to determine  $V_c$  and  $I_{bias}$  (as these circuit parameters don't uniformly impact characteristics across the three models) during training, a decision was made to assign a constant arbitrary value across all  $N_{cla}$  classes. Regarding the Bayes Voting classifier, it offers the ability for tuning. This approach ensures that any notable decrease in hardware accuracy can be traced back to the software  $I_r$  value extraction, simplifying development and minimizing unnecessary complexity. This process is performed once for each specific application, with the resultant parameters then serving as inputs in the classifier. Furthermore, in a complete system implementation, we could export and store the resultant parameters in some form of memory, such as analog memristive type or digital accompanied by low-rate ultra-low-power data converters [61], [62]. By utilizing low-power data converters alongside digital memory, programming the parameter values ( $I_r$ ,  $I_{bias}$ , and  $V_c$ ) during training becomes straightforward.

For every cluster cell, the parameter current  $[I_{ri}]_{i=1}^{N_d}$ , where

$N_d$ , corresponds to the components of the mean vector in the modeled Gaussian probability density function or Euclidean norm. These values can be directly stored in the memory. On the other hand, the parameter voltages  $[V_{ci}]_{i=1}^{N_d}$  regulate the variance of each cluster using a bounded, monotonically increasing, and non-linear function. To establish this function, a single AFC was subjected to simulation with varied  $V_c$  values, and the resulting curves were fitted to a polynomial model (lookup table). This model creates a mapping between acquired variances and excitation voltages  $V_c$ . Additionally, each cluster cell is influenced by  $N_d$  biasing currents  $I_{bias}$ , in Gaussian-based models. These biasing currents result from three distinct parameters: the *prior* probability associated with each class, the cluster weights, and the voting strength for each feature. Crucially, the bias currents were normalized within the range of [3, 9] nA. This normalization ensures proper circuit functionality while keeping power consumption at a low level.

In this research on analog hardware classifiers, the software-based training phase was executed using Python, complemented by an array of indispensable libraries. These included *scipy.io* for efficient loading and saving of *MATLAB* files, *matplotlib* for versatile data visualization, and *pandas* for seamless data manipulation and analysis. The inclusion of *NumPy* bolstered our numerical computing capabilities, facilitating streamlined data handling and processing. Leveraging the rich functionality of *scikit-learn*, we employed various machine learning tools such as Gaussian mixture models, K-means clustering, and classification metrics like classification reports and confusion matrices. Additionally, the *PyTorch* library was instrumental in implementing and training neural network-based classifiers, offering flexibility and scalability in model architecture design. This comprehensive software toolkit enabled robust experimentation and fine-tuning of our classifiers prior to hardware testing, laying a solid foundation for our research endeavors. Regarding the number of epochs, only for LVQ is a hyperparameter which for this specific task is equal to one.

## B. Architecture parameterization

The proposed classifier introduces the ability of post-layout adjustments in terms of dimensionality and classes. This adaptation transforms the operational dynamics of the architecture. This capability complements the tunability inherent in the provided curve (Gaussian or Euclidean), concerning the tuning of  $I_r$ ,  $V_c$ , and  $I_{bias}$  values, as outlined in the previous subsection. This flexibility can be particularly advantageous, in scenarios where a smaller classification task with fewer classes  $n_{cla}$  and features  $n_d$  is required. By designing a larger system comprising  $N_{cla} > n_{cla}$  classes and  $N_d > n_d$  sequentially linked AFCs, the same classifier topology can effectively tackle this reduced classification problem, eliminating the need for a costly fabrication repetitions.

Turning to the tunability of input dimensionality, control is exerted through parameters  $I_r$  and  $V_c$ , alongside input current  $I_{in}$ . For the initial  $(N_d - n_d)$  out of the  $N_d$ , all three parameters  $I_r$ ,  $V_c$ ,  $I_{in}$ , are set to their lowest values, forming a zero current buffer for a specific bias current  $I_{bias} = 0$

(only for the specific AFCs). This configuration channels the system's  $n_d$ -dimensional input to the remaining  $n_d$  AFCs, thereby forming a lower-dimensional classifier. Importantly, reducing dimensions does not proportionally lower the total power consumption, because all the fabricated  $N_d$  AFCs remain almost operational.

The adjustability of the classes is predominantly achieved through the bias current  $I_{bias}$  for hardware classifiers. First, by tuning each AFC using different  $I_{bias}$  value, a weighted approach can be achieved by assigning weights to each feature of the classification problem. On the other hand, by setting the bias current of an inactive feature to zero, it becomes dormant, as its output current drops significantly below  $\ln A$ . This manipulation effectively alters the effective class count. Alternatively, the same goal is achievable by symmetrically configuring  $I_{in}$  and  $I_r$  for each  $n_d$  bump in the inactive class. For instance, all  $I_r$  values can be set to low current values and  $I_{in}$  to high. In contrast to dimension reduction, decreasing class numbers notably curtails the total power consumption, as completely unused fabricated input dimensions are involved. Class tunability becomes particularly advantageous when the hardware classifier is unnecessary. Setting  $I_{bias}$ ,  $I_{in}$  and  $I_r$  for all classes to zero deactivates the classifier, ushering in a power-saving mode.

Finally, the system's adjustability extends to the customization of  $I_{bias}$  and  $V_c$ . Although the bias current and variance voltage values for all AFCs remain constant owing to algorithmic constraints linked to the software implementation and the utilization of AFCs primarily as a comparable distance metric, the classifier's design process maintains flexibility. This enables a potential change in the approach if a more advanced method is developed. Consequently, the design procedure and classifier validation remain unbounded, accommodating fine-tuning adjustments despite the potential early fabrication needs for specific applications.

## V. APPLICATION EXAMPLE AND SIMULATION RESULTS

In this section, we evaluate the proposed circuit architecture described in Section III by applying it to a distinct biomedical classification task. This architecture incorporates three essential mathematical classification models, Bayes, Centroid, and LVQ. By implementing the techniques elucidated in the previous section, a system layout is created capable of handling datasets characterized by  $N_d = 49$  features and  $N_{cla} = 3$  classes [41]. The selection of specific values ( $N_d = 49$  and  $N_{cla} = 3$ ) was made to ensure an area-efficient implementation, aligning with the number of dimensions in the diabetes disease management and prediction dataset file which will be analyzed. To be more specific, within this layout, each of the three classes comprises 49 Gaussian function circuits, 49 Mahalanobis distance implementation circuits, and two primary switches (allowing the selection of either Gaussian or Euclidean approximation). Consequently, this single layout allows us to test all three primary models. Furthermore, this layout's flexibility enables adjustments to accommodate varying numbers of classes (up to three) and dimensionalities (up to 49). The layout implementation, shown in Fig. 10. In

order to mitigate the mismatch effects due to process and fabrication variations the following measures were taken at the layout level

- Each mirror was constructed using the common centroid technique with multiple instances for each device and the reference device is placed in the center of the transistor array forming the mirror.
- Each differential pair is constructed interleaving the transistors that form them.
- Dummy devices are added around the devices that need to be matched.

To further reduce the mismatch between the various devices that need to be matched (i.e. mirrors and differential pairs), their gate length is selected to be much greater than the process minimum allowable value [63]. Compared to related cascaded classifiers [15]–[17], [19], [20], [24], [26]–[28] each AFC includes an additional CM, offering the capability for independent operation at the expense of an increased area. The total area is equal to  $0.7539mm^2$  (the layout incorporates the three analog classifiers as proposed).

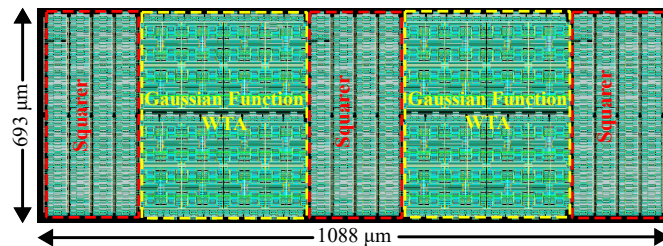


Fig. 10: Layout of the proposed classifier architecture. The total area is equal to  $0.7539mm^2$ . Common-centroid technique is used to address manufacturing considerations.

The dataset [41], diabetes disease management and prediction, from the UCI Machine Learning Repository, offers a comprehensive repository of information regarding diabetes-related healthcare in the United States for the years spanning from 1999 to 2008. It contains  $N_d = 49$  features,  $N_{cla} = 3$  classes. It provides a wealth of data encompassing patient demographics, admission and discharge details, medical diagnoses, and treatment procedures across 130 different hospitals. This dataset serves as a valuable resource for healthcare researchers, policymakers, and data analysts interested in understanding diabetes trends, treatment outcomes, and healthcare disparities over a decade. Researchers can leverage this dataset to investigate factors influencing diabetes management, develop predictive models, and contribute to the enhancement of healthcare strategies aimed at improving diabetes care and outcomes in the United States. It represents a crucial tool in advancing our knowledge of diabetes-related healthcare and guiding evidence-based decisions to enhance patient well-being.

The concept here is that a low-power analog integrated system is designed for diabetes prediction. In this scenario, let's consider a person who has been diagnosed with diabetes or is at risk of developing it. This individual wants to monitor their blood glucose levels continuously to manage their condition effectively and minimize the risk of complications. The



low-power analog integrated system could consist of wearable sensors that continuously monitor the person's physiological parameters, such as blood glucose levels, heart rate variability, and other relevant biometric data. These sensors are designed to be lightweight, unobtrusive, and energy-efficient, allowing the person to wear them comfortably throughout the day without frequent recharging. For example, an integrated glucose sensor consumes less than  $100\mu W$  [64], and the proposed analog classifier consumes less than  $700nW$ . In contrast, digital counterparts consume between  $150\mu W$  to  $10mW$  since they require power-hungry conversions from the analog to the digital domain [65]. As a result, the total power consumption is reduced multiple times.

The system incorporates low-power analog signal processing techniques to accurately capture and process the raw physiological signals from the sensors. Once the physiological data is captured and processed, the system employs machine learning algorithms in hardware to analyze the data and predict the likelihood of blood glucose fluctuations or diabetic events, such as hypoglycemia (low blood sugar) or hyperglycemia (high blood sugar). These algorithms could be trained on large datasets of historical physiological data collected from individuals with diabetes, enabling the system to learn patterns and correlations indicative of impending glucose fluctuations. The predictions generated by the system are then relayed to the user in a dedicated wearable device. This allows the user to take proactive measures. Overall, the low-power analog integrated system for diabetes prediction offers a non-invasive, continuous monitoring solution that empowers individuals with diabetes to proactively manage their condition and lead healthier lives.

To demonstrate the efficacy of the proposed classification pipeline, we conduct a rigorous training and validation process. This procedure was repeated 20 times for the dataset to ensure robust classification accuracy and minimize the influence of random factors associated with software-based train-test splits. In each iteration, we compare both the analog and software implementations using identical training and validation datasets. The accuracy rate for each iteration is presented in the following figures, and the best, average, and worst accuracy values are summarized in the Table II. Additionally, a sensitivity analysis of the circuit is performed through Monte Carlo simulations (over process and mismatch variations), in the dataset, involving  $N = 100$  data points. Finally, we assess the performance of the proposed classifier, which is designed based on the aforementioned validation procedure, using the implemented layout in post-layout simulations. It's worth noting that all the circuits and layouts discussed in this study were developed within the Cadence IC design suite, employing TSMC's 90nm CMOS process.

The fixed mean value, variance, and height for the activation functions were initially determined based on independent data that were not used in the next sets (validation set). Subsequently, independent data were employed, of which 70% were selected each iteration, related to the training of the proposed classifier and the remaining 30% were used to calculate the classification accuracy, as presented in the simulation results. This split is employed to train a software-based classifier. To

ensure a fair and unbiased comparison of results, both the software-based and hardware-based implementations of the classifier are evaluated on the same test set. Furthermore, in order to mitigate the impact of random variation stemming from the train-test split, the entire training and validation process is repeated 20 times. Table II presents a summary of the outcomes pertaining to the management and prediction of diabetes disease across all the classifiers that have been implemented. Furthermore, for illustrative purposes, Figs. 11, 12, and 13 provide visual representations of these relevant results. Note that there is a slight decrease in hardware accuracy compared to software. This is primarily because the circuits generate an approximation of the requested functions (Gaussian and Mahalanobis) but are not ideal. However, the training of the parameters forms the foundation of their ideal model.

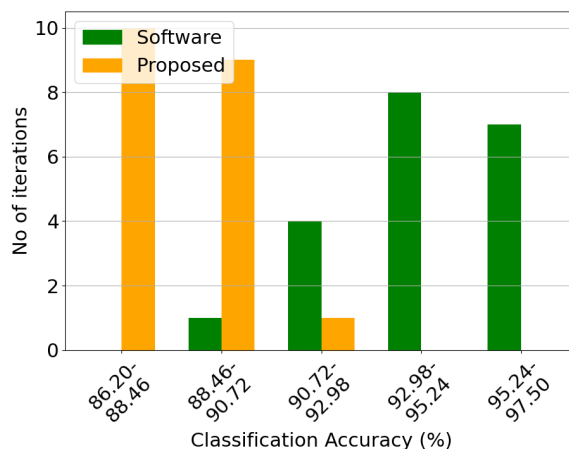


Fig. 11: Classification results of the Bayes model and the equivalent software model on the diabetes disease management and prediction dataset over 20 iterations.

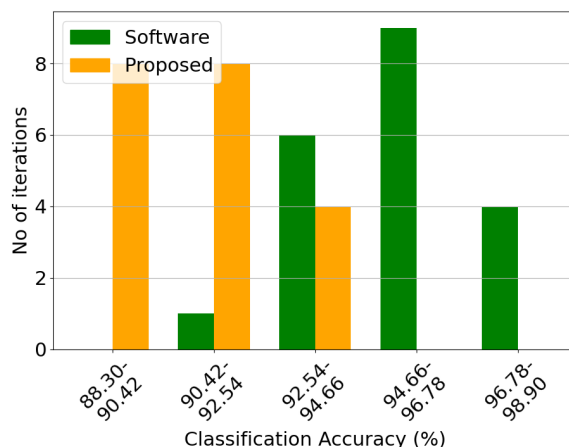


Fig. 12: Classification results of the Centroid model and the equivalent software model on the diabetes disease management and prediction dataset over 20 iterations.

The classifier architecture's robustness is further assessed by conducting a sensitivity analysis through Monte-Carlo simulations. In this evaluation, 100 runs are executed using the parameters and the test set from one of the previous

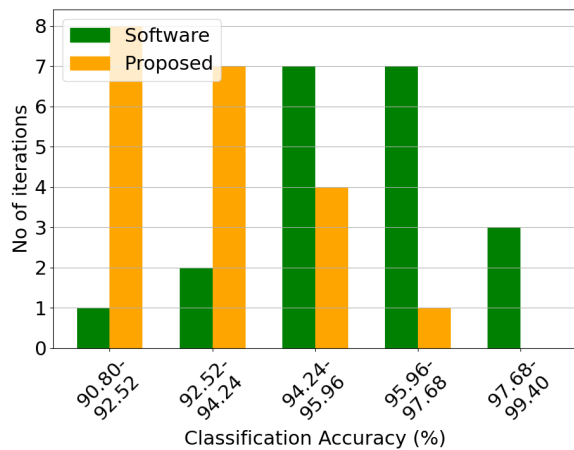


Fig. 13: Classification results of the LVQ model and the equivalent software model on the diabetes disease management and prediction dataset over 20 iterations.

TABLE II: Classification results on the diabetes disease management and prediction dataset over 20 iterations.

	Best (%)	Worst (%)	Mean (%)
Bayes Software	97.50	90.50	94.18
Bayes Hardware	91.20	86.20	88.45
Centroid Software	98.90	91.70	95.49
Centroid Hardware	94.00	88.30	90.90
LVQ Software	99.40	92.00	95.90
LVQ Hardware	96.00	90.80	93.22

20 iterations. The resulting Monte-Carlo analysis histograms are depicted in Fig. 14. The mean value (range:  $\mu_M = 89.20 - 93.58\%$  depending on the mathematical model) and the standard deviation (range:  $\sigma_M = 0.83 - 1.11\%$  depending on the mathematical model) from this analysis signify strong sensitivity characteristics and robust operational performance.

Except for Monte-Carlo analysis, the proposed classifiers are tested over Process-Voltage-Temperature (PVT) variations. The selected corners are TT, SS, FF, SF, FS (T:Typical, S:Slow, F:Fast). Additionally, the power supply rails variation is set in the range  $V_{DD} = -V_{SS} = 0.25V$  to  $V_{DD} = -V_{SS} = 0.35V$ . Concerning temperature, the selected range is from  $-25^\circ C$  to  $125^\circ C$ . The same PVT variations with corners are used for the Monte Carlo analysis. All three classifiers demonstrate robustness across corners, achieving a minimum classification accuracy of 83.78%, 84.41% and 86.02% for Bayes, Centroid and LVQ models respectively (worst case scenario). The worst case corner is SS,  $-25^\circ C$ ,  $V_{DD} = -V_{SS} = 0.25V$  and low software-based accuracy. The robustness is achieved using cascode current mirrors and long (L) transistors. Also, simulation indicates that the insensitivity of the output results to the biasing parameters ( $I_r$ ,  $I_{bias}$  and  $V_c$ ) as long as they remain within the acceptable operating range. Finally, corner and Monte-Carlo simulations indicate that the circuit performance is sufficiently robust, and calibration is not necessary as long as the parameter set remain within their acceptable ranges.

In order to provide a more complete approach, we also calculate the metric Sensitivity which is one of the main characteristic for the diabetes disease management and prediction dataset. This metric is related to the true predicted

diabetes cases over the total diabetes cases (the total cases combine the predicted cases and the missed cases). The mean specificity values for Bayes, Centroid and LVQ models are 96.73%, 97.18%, and 98.58%, respectively.

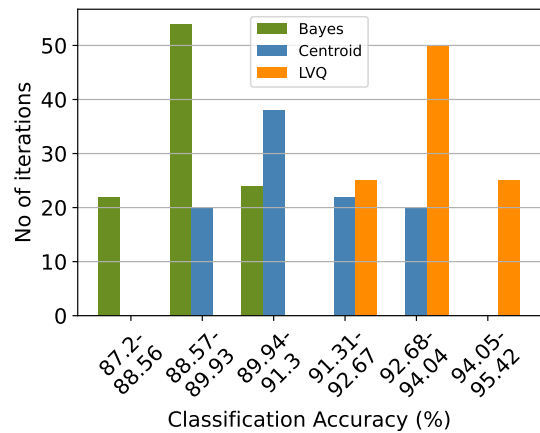


Fig. 14: Post-layout Monte-Carlo simulation results of the three classification models on the diabetes disease management and prediction dataset with  $\mu_M = 89.20\%$  and  $\sigma_M = 0.83\%$ ,  $\mu_M = 91.19\%$  and  $\sigma_M = 1.11\%$  and  $\mu_M = 93.58\%$  and  $\sigma_M = 0.98\%$  for the Bayes, Centroid and LVQ respectively.

## VI. DISCUSSION AND COMPARISON

Within the existing literature, it is noteworthy that the majority of analog classifiers are typically tailored for specific applications. This situation hinders the feasibility of conducting an impartial comparison among diverse implementations. Consequently, this allows us to adapt the design of related classifiers to fit the same application, facilitating an overall performance comparison both between ML models and different approaches. All the summarized classifiers are implemented in a TSMC 90 nm CMOS process technology, with power supply rails selected based on the operating region and a trade-off between higher accuracy and lower power consumption. All are implemented for the same dataset (diabetes disease management). In particular, Table III provides a performance overview of this research alongside related classifiers (referred in Introduction), within the context of diabetes disease management and prediction. The dataset used here has 49 features. Some architectures (multivariate activation functions) in Table III e.g. [15] cannot be extended to have so many inputs since their functionality is reduced dramatically. To this end all such architectures have been modified to 14 input features. To be able to feed them with the dataset as unbiased as possible, we used PCA to convert the feature vector dimensionality from 49 (originally) to 14.

Specifically, the implementations of ML models referenced in Table III rely on approximations of equivalent mathematical models. Previous research [15]–[17], [19], [20], [24], [26]–[28] builds upon the concept of multivariate activation functions, which are implemented in analog circuitry using cascaded AFCs, where the bias current of each circuit is the output current of the previous stage. For example, the output current ( $I_{out}$ ) of the Gaussian function circuit reaches

TABLE III: Analog classifiers' comparison on the diabetes disease management and prediction dataset. \*The dataset's features' set has been reduced using PCA to match the input dimensionality of these classifiers

	Classifier	Best accuracy (%)	Worst accuracy (%)	Mean accuracy (%)	Power consumption ( $\mu W$ )	Processing speed ( $\frac{\text{classifications}}{s}$ )	Energy (pJ) per classification	No. of Dimensions
<b>This work</b>	Bayes	94.40	89.70	91.54	0.688	320K	2.15	49
<b>This work</b>	Centroid	97.20	90.90	94.21	0.572	380K	1.51	49
<b>This work</b>	LVQ	99.20	94.10	96.48	0.617	275K	2.24	49
[15]	Bayes*	81.70	72.70	77.43	0.988	170K	5.81	14
[16]	GMM*	82.40	74.30	77.83	2.43	170K	14.29	14
[17]	RBF*	82.80	73.90	78.51	29.54	170K	174.35	14
[18]	RBF-NN	93.10	82.30	87.93	1.63	270K	6.04	49
[19]	SVM*	87.80	82.30	84.58	973.4	870K	1120	14
[20]	SVM*	85.70	81.60	83.48	71.8	140K	512.86	14
[21]	MLP	95.20	92.30	93.83	1120	930K	1200	49
[22]	K-means	97.30	88.70	91.22	355.7	5M	71.14	49
[23]	SVR	91.90	86.70	88.65	112.3	870K	129.08	49
[24]	SOM*	95.30	91.10	93.12	891	180K	4950	14
[25]	LSTM	100.00	95.80	98.74	78000	870M	89.65	49
[26]	Fuzzy*	87.80	79.40	83.72	1.21	4.55K	265.93	14
[27]	Threshold*	85.40	78.30	84.31	0.522	100K	5.22	14
[28]	Centroid*	84.70	78.60	81.93	4.54	170K	26.71	14

its maximum value when  $I_{in} = I_r$ . This maximum value is equal to  $I_{bias}$ . For  $I_{in}$  non equal to  $I_r$ , it leads to a degradation of the current from the input to the output of the multivariate Gaussian function circuit. Due to this limitation, the maximum number of cascaded circuits in practical applications is typically less than 20. Potential resolutions to this issue include augmenting the bias current to ensure the proper functioning of all AFCs or employing the PCA technique. The aforementioned solution is constrained by the maximum allowable bias current, as it necessitates all transistors to operate within the sub-threshold region. Also, in the related works the desired operating range is limited. When the parameter selected from the training is close to the power supply rails, there is a reduction in the output current value compared to another parameter located at the center of the power supply. Consequently, the output current of the AFC may reduce below the permissible operating current for subsequent AFCs.

The novelty of the proposed work lies in addressing the aforementioned operational issues. To achieve this, the AFCs are not connected in cascade; instead, their output currents are summed on the output node of each class. Consequently, the bias ( $I_{bias}$ ), parameter ( $I_r$ ) and input ( $I_{in}$ ) currents of each AFC can be adjusted as low as necessary just to ensure proper circuit operation. This approach effectively addresses

the limitation introduced by cascade circuits, enabling efficient handling of multiple input features while maintaining exceptional levels of accuracy and power consumption. Moreover, it provides an additional degree of freedom, as the weights of each feature can be tuned independently, based on software implementation.

This work outperforms related analog classifiers in power consumption and energy per classification. It's crucial to highlight that, in this particular application, a high number of input dimensions is encountered by the classifiers. The proposed topology provides a notable advantage by eliminating the need for PCA [30], allowing us to utilize all 49 input dimensions without any loss of information. In contrast, other related topologies, that leverage cascaded AFCs, must reduce the dimensions to 14 to achieve optimal accuracy, which represents a significant limitation in these works. While our network exhibits the capability to accurately classify all 3 classes, we have transformed the problem into a binary classification scenario for a more meaningful comparison with binary analog classifiers [19], [20], [23], [24], [26], [27]. As a result, the performance of the proposed architecture for the binary-classification problem is superior to the 3-class task for the same dataset.

Regarding the area, the most efficient implementation among the previously compared is [27] with  $0.101mm^2$  for

the diabetes disease management dataset (PCA has reduced the number of dimensions at the cost of accuracy). The related classifiers [15]–[17], [19], [20], [24], [26], [28] which build upon the concept of multivariate activation functions occupy chip area varying in the range  $0.207mm^2$ – $1.271mm^2$  (PCA has reduced the number of dimensions at the cost of accuracy). Finally, the most complicated works [18], [21]–[23], [25] occupy chip area varying in the range  $1.174mm^2$ – $6.100mm^2$ . All are implemented for the diabetes dataset.

Among the ML models implemented in this work, i.e. Bayes, Centroid and LVQ, the last one seems to achieve the best classification accuracy. This is due to the complexity of the LVQ algorithm over the other approaches. Moreover, this implementation outperforms in terms of accuracy all the other classifiers presented in Table III, except for the LSTM algorithm, since the latter is advantageous in the context of model complexity and hardware-approximation efficiency. It has to be noted that the above performance is achieved with the minimum energy per classification in comparison to the other approaches. The minimum power consumption is attained by the Threshold classifier, although sacrificing accuracy and processing speed due to its model's simplicity. It is important to highlight that, in this kind of biomedical applications, rapid processing speed is not a vital specification, because of their low frequency [66]. For this reason, in the proposed work, processing speed is intentionally diminished in order to achieve better accuracy and power consumption.

Another point - In real time systems, increased sampling rates results in increased data points, that offers a much richer data than that is often reported in classification datasets used for benchmarking. This means, even if accuracy prediction per sample is lower, overall accuracy of the system can be higher by using voting schemes in conjunction with multi classifier systems.

## VII. CONCLUSION

This work introduced an analog integrated architecture implementing a Voting classification algorithm, targeting low power applications, capable of accommodating several features and achieving good accuracy (more than 86.20%). To evaluate the proposed architecture, a thorough comparison was conducted against other analog classifiers in the literature, using real-world biomedical dataset. All implementations were power efficient (less than  $688nW$ ) and low supply voltage (only 0.6V). The model training was performed using software-based classifiers. The robustness of the proposed architecture, with respect to circuit component variations, was evaluated using Monte Carlo and Corner simulation. Furthermore, the simulation also demonstrates robustness towards circuit biasing parameters. The hardware design, schematic and post-layout simulation was performed in the Cadence IC Suite, employing TSMC's 90 nm CMOS process technology.

## REFERENCES

[1] S. Harrer, P. Shah, B. Antony, and J. Hu, "Artificial intelligence for clinical trial design," *Trends in pharmacological sciences*, vol. 40, no. 8, pp. 577–591, 2019.

[2] I. Zafar, S. Anwar, W. Yousaf, F. U. Nisa, T. Kausar, Q. ul Ain, A. Unar, M. A. Kamal, S. Rashid, K. A. Khan *et al.*, "Reviewing methods of deep learning for intelligent healthcare systems in genomics and biomedicine," *Biomedical Signal Processing and Control*, vol. 86, p. 105263, 2023.

[3] F. Boniolo, E. Dorigatti, A. J. Ohnmacht, D. Saur, B. Schubert, and M. P. Menden, "Artificial intelligence in early drug discovery enabling precision medicine," *Expert Opinion on Drug Discovery*, vol. 16, no. 9, pp. 991–1007, 2021.

[4] A. N. Navaz, M. A. Serhani, H. T. El Kassabi, N. Al-Qirim, and H. Ismail, "Trends, technologies, and key challenges in smart and connected healthcare," *Ieee Access*, vol. 9, pp. 74 044–74 067, 2021.

[5] M. Alshamrani, "Iot and artificial intelligence implementations for remote healthcare monitoring systems: A survey," *Journal of King Saud University-Computer and Information Sciences*, vol. 34, no. 8, pp. 4687–4701, 2022.

[6] Y. Wei, J. Zhou, Y. Wang, Y. Liu, Q. Liu, J. Luo, C. Wang, F. Ren, and L. Huang, "A review of algorithm & hardware design for ai-based biomedical applications," *IEEE transactions on biomedical circuits and systems*, vol. 14, no. 2, pp. 145–163, 2020.

[7] J. P. Lynch, H. Sohn, and M. L. Wang, *Sensor technologies for civil infrastructures: Volume 1: Sensing hardware and data collection methods for performance assessment*. Woodhead Publishing, 2022.

[8] M. Sahu, R. Gupta, R. K. Ambasta, and P. Kumar, "Artificial intelligence and machine learning in precision medicine: A paradigm shift in big data analysis," *Progress in Molecular Biology and Translational Science*, vol. 190, no. 1, pp. 57–100, 2022.

[9] N. Mirchandani, Y. Zhang, S. Abdelfattah, M. Onabajo, and A. Shrivastava, "Modeling and simulation of circuit-level nonidealities for an analog computing design approach with application to eeg feature extraction," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 42, no. 1, pp. 229–242, 2022.

[10] W. Haensch, T. Gokmen, and R. Puri, "The next generation of deep learning hardware: Analog computing," *Proceedings of the IEEE*, vol. 107, no. 1, pp. 108–122, 2018.

[11] F. Liu, S. Deswal, A. Christou, Y. Sandamirskaya, M. Kaboli, and R. Dahiya, "Neuro-inspired electronic skin for robots," *Science robotics*, vol. 7, no. 67, p. eabl7344, 2022.

[12] Y.-T. Hsieh, K. Anjum, and D. Pompili, "Ultra-low power analog recurrent neural network design approximation for wireless health monitoring," in *2022 IEEE 19th International Conference on Mobile Ad Hoc and Smart Systems (MASS)*. IEEE, 2022, pp. 211–219.

[13] M. Elbtity, A. Singh, B. Reidy, X. Guo, and R. Zand, "An in-memory analog computing co-processor for energy-efficient cnn inference on mobile devices," in *2021 IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*. IEEE, 2021, pp. 188–193.

[14] T. Zeng, J. Xie, Y. Zhou, F. Fan, and S. Wen, "The reflective optical analog computing system based on cholesteric liquid," *Adv. Mater*, vol. 31, p. 1806172, 2019.

[15] V. Alimisis, G. Gennis, C. Dimas, and P. P. Sotiriadis, "An analog bayesian classifier implementation, for thyroid disease detection, based on a low-power, current-mode gaussian function circuit," in *2021 International conference on microelectronics (ICM)*. IEEE, 2021, pp. 153–156.

[16] V. Alimisis, G. Gennis, K. Touloupas, C. Dimas, M. Gourdouparis, and P. P. Sotiriadis, "Gaussian mixture model classifier analog integrated low-power implementation with applications in fault management detection," *Microelectronics Journal*, vol. 126, p. 105510, 2022.

[17] S.-Y. Peng, P. E. Hasler, and D. V. Anderson, "An analog programmable multidimensional radial basis function based classifier," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 54, no. 10, pp. 2148–2158, 2007.

[18] A. Reda, L. Qi, Y. Li, and G. Wang, "A generic nano-watt power fully tunable 1-d gaussian kernel circuit for artificial neural network," *IEEE Trans. Circuits Syst. II Express Briefs*, vol. 67, p. 3008679, 2020.

[19] K. Kang and T. Shibata, "An on-chip-trainable gaussian-kernel analog support vector machine," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 57, no. 7, pp. 1513–1524, 2009.

[20] V. Alimisis, G. Gennis, M. Gourdouparis, C. Dimas, and P. P. Sotiriadis, "A low-power analog integrated implementation of the support vector machine algorithm with on-chip learning tested on a bearing fault application," *Sensors*, vol. 23, no. 8, p. 3978, 2023.

[21] K. Lee, J. Park, and H.-J. Yoo, "A low-power, mixed-mode neural network classifier for robust scene classification," *Journal of Semiconductor Technology and Science*, vol. 19, no. 1, pp. 129–136, 2019.

- [22] R. Zhang and T. Shibata, "An analog on-line-learning k-means processor employing fully parallel self-converging circuitry," *Analog Integrated Circuits and Signal Processing*, vol. 75, pp. 267–277, 2013.
- [23] R. Zhang, N. Uetake, T. Nakada, and Y. Nakashima, "Design of programmable analog calculation unit by implementing support vector regression for approximate computing," *IEEE Micro*, vol. 38, no. 6, pp. 73–82, 2018.
- [24] F. Li, C.-H. Chang, and L. Siek, "A compact current mode neuron circuit with gaussian taper learning capability," in *2009 IEEE international symposium on circuits and systems*. IEEE, 2009, pp. 2129–2132.
- [25] Z. Zhao, A. Srivastava, L. Peng, and Q. Chen, "Long short-term memory network design for analog computing," *ACM Journal on Emerging Technologies in Computing Systems (JETC)*, vol. 15, no. 1, pp. 1–27, 2019.
- [26] E. Georgakilas, V. Alimisis, G. Gennis, C. Aletraris, C. Dimas, and P. P. Sotiriadis, "An ultra-low power fully-programmable analog general purpose type-2 fuzzy inference system," *AEU-International Journal of Electronics and Communications*, vol. 170, p. 154824, 2023.
- [27] V. Alimisis, G. Gennis, E. Tsouvalas, C. Dimas, and P. P. Sotiriadis, "An analog, low-power threshold classifier tested on a bank note authentication dataset," in *2022 International Conference on Microelectronics (ICM)*. IEEE, 2022, pp. 66–69.
- [28] V. Alimisis, V. Mouzakis, G. Gennis, E. Tsouvalas, and P. P. Sotiriadis, "An analog nearest class with multiple centroids classifier implementation, for depth of anesthesia monitoring," in *2022 International Conference on Smart Systems and Power Management (IC2SPM)*. IEEE, 2022, pp. 176–181.
- [29] V. Alimisis, M. Gourdouparis, G. Gennis, C. Dimas, and P. P. Sotiriadis, "Analog gaussian function circuit: Architectures, operating principles and applications," *Electronics*, vol. 10, no. 20, p. 2530, 2021.
- [30] T. Davies and T. Fearn, "Back to basics: the principles of principal component analysis," *Spectroscopy Europe*, vol. 16, no. 6, p. 20, 2004.
- [31] F. Hu, S. Lakdawala, Q. Hao, and M. Qiu, "Low-power, intelligent sensor hardware interface for medical data preprocessing," *IEEE Transactions on Information Technology in Biomedicine*, vol. 13, no. 4, pp. 656–663, 2009.
- [32] C. Bachmann, M. Ashouei, V. Pop, M. Vidojkovic, H. De Groot, and B. Gyselinckx, "Low-power wireless sensor nodes for ubiquitous long-term biomedical signal monitoring," *IEEE Communications Magazine*, vol. 50, no. 1, pp. 20–27, 2012.
- [33] Y. Zhang, N. Mirchandani, M. Onabajo, and A. Shrivastava, "Rssi amplifier design for a feature extraction technique to detect seizures with analog computing," in *2020 IEEE international symposium on circuits and systems (ISCAS)*. IEEE, 2020, pp. 1–5.
- [34] M. Yang, C.-H. Yeh, Y. Zhou, J. P. Cerqueira, A. A. Lazar, and M. Seok, "A  $1\mu\text{w}$  voice activity detector using analog feature extraction and digital deep neural network," in *2018 IEEE International Solid-State Circuits Conference (ISSCC)*. IEEE, 2018, pp. 346–348.
- [35] M. Yang, H. Liu, W. Shan, J. Zhang, I. Kiselev, S. J. Kim, C. Enz, and M. Seok, "Nanowatt acoustic inference sensing exploiting nonlinear analog feature extraction," *IEEE Journal of Solid-State Circuits*, vol. 56, no. 10, pp. 3123–3133, 2021.
- [36] S. Ray and P. R. Kinget, "Ultra-low-power and compact-area analog audio feature extraction based on time-mode analog filterbank interpolation and time-mode analog rectification," *IEEE Journal of Solid-State Circuits*, vol. 58, no. 4, pp. 1025–1036, 2022.
- [37] R. N. Kandala, R. Dhuli, P. Pławiak, G. R. Naik, H. Moeinzadeh, G. D. Gargiulo, and S. Gunnam, "Towards real-time heartbeat classification: evaluation of nonlinear morphological features and voting method," *Sensors*, vol. 19, no. 23, p. 5079, 2019.
- [38] S. Kumari, D. Kumar, and M. Mittal, "An ensemble approach for classification and prediction of diabetes mellitus using soft voting classifier," *International Journal of Cognitive Computing in Engineering*, vol. 2, pp. 40–46, 2021.
- [39] A. Yousaf, M. Umer, S. Sadiq, S. Ullah, S. Mirjalili, V. Rupapara, and M. Nappi, "Emotion recognition by textual tweets classification using voting classifier (lr-sgd)," *IEEE Access*, vol. 9, pp. 6286–6295, 2020.
- [40] E.-S. M. El-Kenawy, A. Ibrahim, S. Mirjalili, M. M. Eid, and S. E. Hussein, "Novel feature selection and voting classifier algorithms for covid-19 classification in ct images," *IEEE access*, vol. 8, pp. 179 317–179 335, 2020.
- [41] C.-K. D. J. Clore, John and B. Strack, "Diabetes 130-US hospitals for years 1999-2008," UCI Machine Learning Repository, 2014, DOI: <https://doi.org/10.24432/C5230J>.
- [42] R. Delgado, "A semi-hard voting combiner scheme to ensemble multi-class probabilistic classifiers," *Applied Intelligence*, vol. 52, no. 4, pp. 3653–3677, 2022.
- [43] J. Cao, S. Kwong, R. Wang, X. Li, K. Li, and X. Kong, "Class-specific soft voting based multiple extreme learning machines ensemble," *Neurocomputing*, vol. 149, pp. 275–284, 2015.
- [44] M. Wiggins, A. Saad, B. Litt, and G. Vachtsevanos, "Evolving a bayesian classifier for ecg-based age classification in medical applications," *Applied soft computing*, vol. 8, no. 1, pp. 599–608, 2008.
- [45] D. M. Diab and K. M. El Hindi, "Using differential evolution for fine tuning naïve bayesian classifiers and its application for text classification," *Applied Soft Computing*, vol. 54, pp. 183–199, 2017.
- [46] C. M. Bishop and N. M. Nasrabadi, *Pattern recognition and machine learning*. Springer, 2006, vol. 4, no. 4.
- [47] S. Taheri and M. Mammadov, "Learning the naive bayes classifier with optimization models," *International Journal of Applied Mathematics and Computer Science*, vol. 23, no. 4, pp. 787–795, 2013.
- [48] C. Liu, W. Wang, G. Tu, Y. Xiang, S. Wang, and F. Lv, "A new centroid-based classification model for text categorization," *Knowledge-Based Systems*, vol. 136, pp. 15–26, 2017.
- [49] C. Manning, P. Raghavan, and H. Schütze, "Vector space classification," *Introduction to Information Retrieval*, pp. 289–317, 2008.
- [50] H. Park, M. Jeon, and J. B. Rosen, "Lower dimensional representation of text data based on centroids and least squares," *BIT Numerical mathematics*, vol. 43, pp. 427–448, 2003.
- [51] R. De Maesschalck, D. Jouan-Rimbaud, and D. L. Massart, "The mahalanobis distance," *Chemometrics and intelligent laboratory systems*, vol. 50, no. 1, pp. 1–18, 2000.
- [52] T. Kohonen, "Self-organizing maps, ser," *Information Sciences. Berlin: Springer*, vol. 30, 2001.
- [53] D. Nova and P. A. Estévez, "A review of learning vector quantization classifiers," *Neural Computing and Applications*, vol. 25, pp. 511–524, 2014.
- [54] W.-L. Hung, D.-H. Chen, and M.-S. Yang, "Suppressed fuzzy-soft learning vector quantization for mri segmentation," *Artificial intelligence in medicine*, vol. 52, no. 1, pp. 33–43, 2011.
- [55] N. Pradhan, P. Sadasivan, and G. Arunodaya, "Detection of seizure activity in eeg by an artificial neural network: A preliminary study," *Computers and Biomedical Research*, vol. 29, no. 4, pp. 303–313, 1996.
- [56] J.-Y. Jeng, T.-F. Mau, and S.-M. Leu, "Prediction of laser butt joint welding parameters using back propagation and learning vector quantization networks," *Journal of Materials Processing Technology*, vol. 99, no. 1-3, pp. 207–218, 2000.
- [57] M. Kaden, K. S. Bohnsack, M. Weber, M. Kudła, K. Gutowska, J. Blazewicz, and T. Villmann, "Learning vector quantization as an interpretable classifier for the detection of sars-cov-2 types based on their rna sequences," *Neural Computing and Applications*, vol. 34, no. 1, pp. 67–78, 2022.
- [58] E. Seevinck and R. J. Wiegerink, "Generalized translinear circuit principle," *IEEE journal of solid-state circuits*, vol. 26, no. 8, pp. 1098–1102, 1991.
- [59] J. Lazzaro, S. Ryckebusch, M. A. Mahowald, and C. A. Mead, "Winner-take-all networks of o (n) complexity," *Advances in neural information processing systems*, vol. 1, 1988.
- [60] X. Ying, "An overview of overfitting and its solutions," in *Journal of physics: Conference series*, vol. 1168. IOP Publishing, 2019, p. 022022.
- [61] M. Hock, A. Hartel, J. Schemmel, and K. Meier, "An analog dynamic memory array for neuromorphic hardware," in *2013 European Conference on Circuit Theory and Design (ECCTD)*. IEEE, 2013, pp. 1–4.
- [62] R. Li and H. Fariborzi, "Ultra-low power data converters with beol nem relays," in *2018 IEEE 61st International Midwest Symposium on Circuits and Systems (MWSCAS)*. IEEE, 2018, pp. 627–630.
- [63] A. K. Sharma, M. Madhusudan, S. M. Burns, P. Mukherjee, S. Yaldiz, R. Harjani, and S. S. Sapatnekar, "Common-centroid layouts for analog circuits: Advantages and limitations," in *2021 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, 2021, pp. 1224–1229.
- [64] N. Kazemi, M. Abdolrazzagli, P. E. Light, and P. Musilek, "In-human testing of a non-invasive continuous low-energy microwave glucose sensor with advanced machine learning capabilities," *Biosensors and Bioelectronics*, vol. 241, p. 115668, 2023.
- [65] K. Lata, S. Saini, and G. Sinha, "Vlsi and hardware implementation using machine learning methods: A systematic literature review," *VLSI and Hardware Implementations using Modern Machine Learning Methods*, pp. 1–21, 2021.
- [66] H.-T. Wu, "Current state of nonlinear-type time-frequency analysis and applications to high-frequency biomedical signals," *Current Opinion in Systems Biology*, vol. 23, pp. 8–21, 2020.

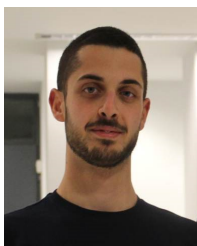


**Vassilis Alimisis** (Student Member IEEE), received the B.Sc. in Physics (top 1%) and the M.Sc. degree in Electronics and Communications from the University of Patras, Greece, in 2017 and 2019 respectively. Currently, he is pursuing Ph.D. degree at the National Technical University of Athens (NTUA), Greece, under the supervision of Professor Paul P. Sotiriadis. His Ph.D. Thesis and research are supported and financed by the E.L.K.E. NTUA Scholarships. He is a Teaching Assistant in undergraduate and graduate courses and supervises Diploma Thesis.

He has authored and co-authored several conference papers and journal articles. His main research interests include analog microelectronic circuits, low power electronics, analog computing and integrated circuit architectures with applications in artificial intelligence and machine learning. He has received the Best Paper Award in the IEEE Int. Conf. on Microelectronics 2020, the Best Paper Award in the IEEE Int. Conf. on Microelectronics 2021, the Best Paper Award (3rd Place) in the IEEE Int. Conf. on Microelectronics 2023, the Best Paper Award in IEEE Symposium on Integrated Circuits and Systems Design (SBCCI) 2021 and the Best Paper Award in the 1st International Conference on Frontiers of Artificial Intelligence, Ethics, and Multidisciplinary Applications in 2023. He regularly reviews for many IEEE transactions and conferences and serves on proposal review panels.



**Charis Aletraris** (Student Member IEEE), received the Diploma degree in Electrical and Computer Engineering from the National Technical University of Athens (NTUA), Greece, in 2023. He has co-authored a journal article. His main research interests include analog microelectronic circuits, ultra-low power electronics, analog computing and integrated circuit architectures with applications in artificial intelligence and machine learning.



**Nikolaos P. Eleftheriou** (Student Member IEEE), is a Senior Graduate Student in the Department of Electrical and Computer Engineering of the National Technical University of Athens (NTUA), Greece. Currently he is pursuing his Diploma Thesis under the supervision of Professor Paul P. Sotiriadis. He is recipient of the Panagiotis Triantafyllidis Scholarship for undergraduate studies. He has co-authored several conference papers and journal articles. He has received the Best Paper Award (3rd Place) in the IEEE Int. Conf. on Microelectronics (ICM), 2023

and the Best Paper Award in the 1st International Conference on Frontiers of Artificial Intelligence, Ethics, and Multidisciplinary Applications in 2023. His main research interests include microelectronic circuit design, analog circuits and analog hardware computing techniques with applications in fuzzy systems, artificial intelligence and machine learning.



**Emmanouil Anastasios Serlis** (Student Member IEEE), received the Diploma degree in Electrical and Computer Engineering from the National Technical University of Athens (NTUA), Greece, in 2023. He has co-authored a journal article. His main research interests include analog microelectronic circuits, ultra-low power electronics, analog computing and integrated circuit architectures with applications in artificial intelligence and machine learning.



**Alex James** (Senior Member, IEEE) received the Ph.D. degree from Griffith University, Queensland, Australia. He is currently a Professor and the Dean (Academic) of the Kerala University of Digital Sciences, Innovation and Technology (aka Digital University Kerala). He is a Professor-in-Charge with Maker Village, the Chief Investigator of the India Innovation Centre for Graphene and CTO for India Graphene Engineering and Innovation Centre. His research interests include AI-neuromorphic systems (software and hardware), VLSI, and image processing.

He is a member of the IEEE CASS TC on Nonlinear Circuits and Systems, the IEEE CTSoc TC on Quantum in Consumer Technology (QCT), TC on Machine Learning, Deep learning and AI in CE (MDA), the IEEE CASS TC on Cellular Nanoscale Networks and Memristor Array Computing (CNN-MAC), and the IEEE CASS SIG on AgriElectronics. He was a member of the IET Vision and Imaging Network. He is also a member of the BCS' Fellows Technical Advisory Group (F-TAG). He was an Editorial Board Member of Information Fusion (2010–2014). He was awarded the IEEE Outstanding Researcher by the IEEE Kerala Section in 2022, the Kairali Scientist Award (Kairali Gaveshana Puraskaram) for Physical Science in 2022, and the Best Associate Editor for TCAS1 in 2021. He was the Founding Chair of the IEEE CASS Kerala Chapter. He has been serving as an Associate Editor for IEEE Access, since 2017, Frontiers in Neuroscience (Neuromorphic Section), since 2022, IEEE Transactions on Circuits and System— I: Regular Papers (2018–2023), and IEEE Open Journal of Circuits and Systems (2022–2023). He is also serving as an Associate Editor-in-Chief for IEEE Open Journal of Circuits and Systems, since 2024, and an Associate Editor for IEEE Transactions on Biomedical Circuits and Systems, since 2024. He is a Life Member of ACM, a Senior Fellow of HEA, a fellow of the British Computer Society (FBCS), and a fellow of IET (FIET).



**Paul P. Sotiriadis** (Fellow IEEE), is a Professor of Electrical and Computer Engineering of the National Technical University of Athens (NTUA), Greece, the Director of the Electronics Laboratory of the NTUA and a governing board member of the Hellenic (National) Space Center of Greece. He runs a team of 25 researchers. He received the Diploma degree in Electrical and Computer Engineering from the NTUA in 1994, the M.S. degree in Electrical Engineering from Stanford University, USA in 1996 and the Ph.D. degree in Electrical Engineering and

Computer Science from the Massachusetts Institute of Technology, USA, in 2002. In 2002, he joined the faculty of the Johns Hopkins University Electrical and Computer Engineering Department and in 2012 he joined the faculty of the Electrical and Computer Engineering Department of the NTUA. He has authored and coauthored more than 200 research publications, most of them in IEEE journals and conferences, holds one patent, and has contributed several chapters to technical books. Prof. Sotiriadis research interests include the design, optimization, and mathematical modeling of analog, mixed-signal and RF integrated and discrete circuits, sensor and instrumentation architectures with emphasis in biomedical instrumentation, advanced RF frequency synthesis, and, the application of machine learning and general AI in the operation as well as the design of electronic circuits. He has received several awards, including the prestigious Guillemin-Cauer Award from the IEEE Circuits and Systems Society in 2012, the Best Paper Award in the 1st International Conference on Frontiers of Artificial Intelligence, Ethics, and Multidisciplinary Applications in 2023, the Best Paper Award (3rd Place) in the IEEE Int. Conf. on Microelectronics (ICM), 2023, the Best Paper Award in the IEEE International Conf. on Microelectronics (ICM), 2021, the Best Paper Award in the IEEE Symposium on Integr. Circ. and Sys. Design (SBCCI), 2021, the Best Paper Award in the IEEE International Conf. on Microelectronics (ICM), 2020, the Best Paper Award in the IEEE International Conf. on Modern Circ. and Sys. Tech. 2019, Best Paper Award in the IEEE International Frequency Control Symposium 2012, Best Paper Award in the IEEE International Symp. on Circuits and Systems 2007 and the IEEE Circuits and Systems Society (CASS) Outstanding Technical Committee Recognition 2022. Also, he has been in the list of the top 2% most influential researchers in the world in 2020, 2022, and 2023. Dr. Sotiriadis is an Associate Editor of the IEEE Sensors Journal and IEEE Open Journal of Circuits and Systems, has served as an Associate Editor of the IEEE Trans. on Circuits and Systems – I (2016-2020) and the IEEE Trans. on Circuits and Systems – II (2005-2010) and has been a member of technical committees of many conferences. He regularly reviews for many IEEE transactions and conferences and serves on proposal review panels.