

Analog-Domain Time-Series Moment Extraction for Low Power Predictive Maintenance Analytics

Ahish Shylendra, Priyesh Shukla, Swarup Bhunia, and Amit Ranjan Trivedi

Abstract—This work presents a novel low-power CMOS implementation for fast statistical feature extraction from time series. Machine learning (ML) models have become standard for time series processing, however, need to rely on a statistical feature extraction stage. Low power statistical feature extraction from time series has received limited attention despite its central role. Addressing this gap, we present a CMOS-based nonparametric statistical feature extraction. We exploit hardware-level opportunities in the analog domain, such as eliminating additions by current outputs and simplifying kernel cells. We also leverage algorithmic opportunities to utilize continuous-domain sample integration to downsample time series without affecting accuracy. Our propositions are experimentally verified using TSMC 65nm test chip and show $17\text{-}75\times$ lower energy than an advanced digital design on various statistical features. While analog processing is susceptible to non-idealities, co-designing the downstream ML model against such non-idealities can retain accuracy to benefit from the analog domain's area/energy efficiency.

Index Terms—Anomaly detection, statistical feature extraction

I. INTRODUCTION

The gaining prominence of internet-of-things (IoT), cyber-physical systems (CPS), and digital twins have created the need for efficient, real-time *on edge* processing of time-series. In addition, machine learning (ML) methods are becoming prominent for time-series analytics. Mainly, spatial-domain ML models such as convolution neural networks (CNN) and auto-encoder (AE), traditionally developed for images, are being increasingly applied on time series. For example, ADEPOS [1] determines time series anomalies by extracting statistical features such as mean, variance, kurtosis, and other higher-order moments over a sweeping window. The statistical feature map is then processed through AE. The feature extraction reduces the downstream ML model. It fuses data from multiple time series by converting it on a spatial map such that streams with mismatching and varying sampling rate can be efficiently combined. Likewise, in [2], similar time-series preprocessing was followed by wavelet transformation.

Although DNN accelerators are vigorously being researched, efficient on-edge time series feature extraction has received only limited attention. Addressing this critical gap, we make the following key contributions in this paper:

- We present efficient circuits for *nonparametric* feature extraction from time-series. A nonparametric model makes minimal assumptions on the underlying data statistics, unlike parametric models. Therefore, our feature extraction modules are generic, especially suited for testbeds where data statistics is obscure or changes frequently.
- Our time-series processing is *lightweight*. Gaussian kernels are implemented in analog mode to leverage current-

mode output, avoiding dedicated adders for kernel integration and enabling parallel processing. We recast the mathematical framework for statistical moment extraction such that the same hardware units can be multiplexed for any moment order. Our processing also exploits continuous domain sample interpolation, allowing downsampling of time series to minimize workload significantly.

- We also *experimentally* verify our propositions using a 65 nm CMOS test-chip. While analog domain processing is susceptible to various non-idealities, using predictive maintenance as a test example, we show that downstream ML models can be trained against such non-idealities to harness the benefits of analog processing.

II. NON-PARAMETRIC STATISTICS ESTIMATION AND HIGHER-ORDER MOMENTS EXTRACTION

A. Non-Parametric Statistics Estimation

Unlike parametric statistical models such as Gaussian or Poisson distribution, a non-parametric model makes fewer assumptions about the underlying statistics of input data. Thereby, non-parametric statistical models are widely applicable, even for the cases where signal stream statistics changes frequently. For nonparametric density estimation, we adopt Kernel Density Estimation (KDE). Using KDE, the density $f(x)$ of a random variable x can be estimated from observed samples x_i as

$$f(x) = \frac{1}{N} \sum_{i=0}^{N-1} k\left(\frac{x - x_i}{h}\right) \quad (1)$$

where $k()$ is a kernel function, h is kernel function width, and N is number of observed samples. For $k()$, symmetric and unimodal functions such as Gaussian or Triangular are suitable [3]. Fig. 1(a) shows an example density estimation.

B. Statistical Moment Extraction from Density Function

A K^{th} order moment of random variable x is computed as $E[x^K] = \int x^K f(x) dx$. Here, $E[]$ is an expectation operator and $f(x)$ is probability density function. To efficiently extract various order moments, the following scheme is pursued. $f(x)$ is estimated nonparametrically using (1). For K^{th} order moment from $f(x)$, a time-sweep $x = g(t)$ is applied to the function $f(x)$ and the function output is integrated. Aiming hardware efficiency as a key objective, our main rationale behind the approach is to reuse the same density estimator for $f(x)$ and integrate its outputs using appropriate time-sweeps $g(t)$ for various statistical order moments. Hence,

$$\int f(g(t)) dt = \int \frac{1}{g'(g^{-1}(x))} f(x) dx. \quad (2)$$

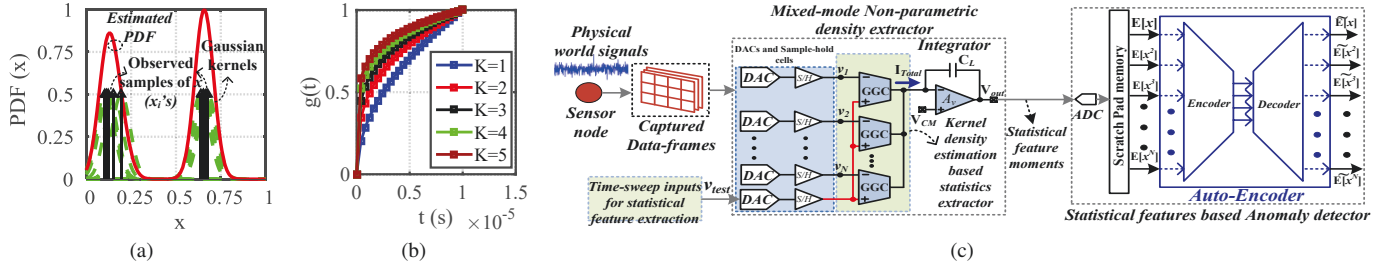


Figure 1: (a) Example nonparametric statistical density estimation. (b) Time-sweeps of $g(t)$ for various order moments. (c) Mixed-mode non-parametric statistical feature extraction from streaming sensor data. Extracted statistical feature map is passed to downstream ML model.

Here, $g' = dg/dt$. Note that by setting $g' = 1/x^K = 1/g^K$, the integral of the above time-sweep will compute the K^{th} order moment of x . The equivalent function for such moment extraction is $g(t) = [(K+1)t]^{1/(K+1)}$. Note that the above relation is valid only when $g(t)$ is invertible, i.e., for each x , $t = g^{-1}(x)$ exists. Since the captured sensor traces are range-limited, the estimated density function is range-limited as well. Considering that sensor outputs are mapped from zero to V_{DD} , $g(t)$ becomes invertible with respect to x , hence the above procedure is applicable.

To extract K^{th} order moment, t should be swept from zero to $\frac{V_{DD}^{K+1}}{(K+1)}$ to consider $0 - V_{DD}$ sweep of x . Practically, to complete time-sweeps for various K within a constant time interval T_{max} , a scaling factor $\alpha_K = \frac{V_{DD}}{[(K+1)T_{max}]^{1/(K+1)}}$ can be applied, i.e. $g(t) = \frac{V_{DD}}{[(K+1)T_{max}]^{1/(K+1)}} [(K+1)t]^{1/(K+1)}$. The true K^{th} order moment can then be extracted by dividing the integral output by α_K . Fig. 1(b) shows such time-sweeps of $g(t)$ for various order moments. At higher order moments, the curvature of $g(t)$ w.r.t. time increases.

III. FRONT-END FOR STATISTICAL FEATURE EXTRACTION

Fig. 1(c) shows a system-level overview of our statistical feature extraction on time-series using the above approach. In Fig. 1(c), a nonparametric density estimator extracts $f(x)$ from the captured time-trace. A feature extractor operates on $f(x)$ to extract various order moments, i.e., $E[x^K]$. Signal moments are digitized and then passed to the following deep learning stage. We discuss each component in detail:

A. Kernel Cell for Density Estimation

We use Gaussian kernel function for our KDE-based density extraction. Fig. 2(a) shows the circuit of Gilbert Gaussian Cell (GGC) to implement a density kernel. An input voltage (corresponding to x in Eq. (1)) is applied to v_{test} terminal and potential applied at v_i is equivalent to x_i in Eq. (1). Fig. 2(b) shows the simulated I_{out} vs. v_{test} characteristics for different v_i using TSMC 65nm models. Notably, peak of the Gaussian I_{out} decreases with increasing common mode voltage v_i . A higher v_i reduces the over-drive voltage of M_8 (tail-device) and reduces the peak current. Degradation in peak current can be overcome by complementing PMOS-based input stage with a similar NMOS-based stage as shown in Fig. 2(c), however doubling the overheads. Even though the analog approach introduces such systematic non-idealities in feature extraction, we will later discuss that downstream ML processing of time-

series can be *co-designed* against such non-idealities.

B. Non-parametric Statistical Feature Extractor

Fig. 2(d) shows mixed-mode implementation of non-parametric statistical feature extractor based on GGCs. KDE is implemented using parallel GGC-based Gaussian kernels. Shorting outputs of GGCs adds-up current from all cells to generate a total current I_{Total} which essentially estimates the likelihood of test input v_{test} against reference samples $[v_1, \dots, v_N]$, i.e., $P(v_{test}|v_1, \dots, v_N)$. To generate the time-sweep of v_{test}^K for the K^{th} order moment, pre-encoded time-sweep values following $v_{test}^K = \frac{V_{DD}}{[(K+1)T_{max}]^{1/(K+1)}} [(K+1)t]^{1/(K+1)}$ are digitally stored and a DAC applies them as input to KDE-based density estimator within a time interval of $0 - T_{max}$. The output of density estimator is integrated using an amplifier to estimate moments in analog domain, which are later digitized for the downstream ML processing. By repeating the time-sweep v_{test}^K for various order moments, the same circuit can be utilized for various moment order extraction.

IV. DENSITY FUNCTION MEASUREMENTS

The test chip for KDE-based density estimation was designed with 65nm TSMC technology. The prototype chip consisted of 10 GGCs and a two-stage OPAMP (operating as transimpedance amplifier), covering a total area of 0.0207 mm^2 . Fig. 3(a) shows the measurement setup. A personal computer (PC) programs Pynq-Z2 FPGA to control the serial-to-parallel (SPI) interface and configure DAC array with input and reference samples. DAC array generates analog reference voltages $[v_1, \dots, v_N]$ and input v_{test} for the density estimator. Subsequently, the test-chip output is read out using a logic analyzer and visualized on PC. Fig. 3(b) shows the output transfer characteristics of on-silicon GGCs at various common-mode voltage. Earlier, we discussed that with only a PMOS-type input stage, GGCs experience a drop in current peak as the common-mode voltage increases. The same is observed experimentally. The measured GGC characteristics were curve-fitted to functionally evaluate the proposed methodology with more GGCs than implemented on the test chip. A key goal of our experimental setup was to analyze the impact of analog-domain non-idealities such as deviation in GGC curves from true Gaussian function. On-silicon KDE consumed an average power of $48 \mu W$ at $V_{DD}=1.2V$ allowing $\sim 350 \text{ MHz}$ processing speed. Since the majority of IoT applications deal with time traces sampled at a much lower frequency, the power

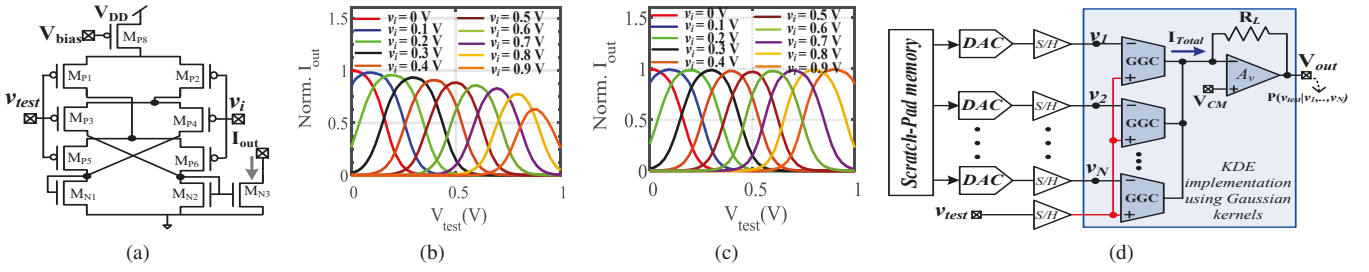


Figure 2: (a) Gaussian kernel implemented using Gilbert Gaussian circuit with PMOS input stage. (b) Transfer characteristics of (a) for different V_i . (c) Transfer characteristics of the Gilbert Gaussian cell with PMOS and NMOS input stage. (d) Mixed-mode non-parametric moment estimation circuit.

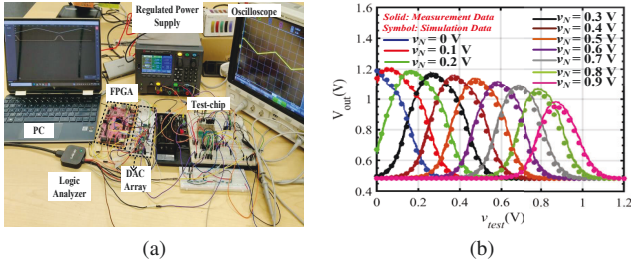


Figure 3: (a) Measurement setup. (b) Measured GGC transfer curves.

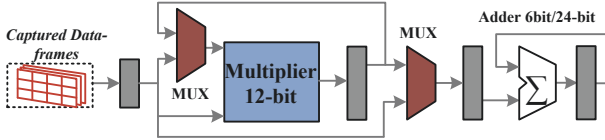


Figure 4: Digital datapath for moment extraction from time series.

dissipation of our setup can be appropriately scaled by bias gating.

Extracting time series moments with non-ideal GCC characteristics in Fig. 3(b), in fact, results in a mixing of various order moments. Since the height of physically-realized Kernel function $k_p(x, x_i)$ depends on the bias point x_i , it can be represented as $k_p(x, x_i) \approx h(x_i) \times k(x, x_i)$ where $k(x, x_i)$ is the true Gaussian-like kernel density function centered at x_i and $h(x_i)$ is a height function. If GGCs are operated with input voltages from zero to 0.8V, $h(x_i)$ decreases from unity with increasing x_i , $h(x_i) \approx 1 - \alpha x_i$. Here, α is the slope of peak height. Consider the K^{th} order moment extraction as

$$E_p[x^K] \approx \sum \int x^k (1 - \alpha x_i) k(x, x_i) dx \quad (3)$$

Under a narrow kernel function width, the above reduces to $E_p(x^K) = E(x^K) - \alpha E(x^{K+1})$, i.e., physically measured K^{th} order moment is a mixture of true K^{th} and $K+1^{th}$ order moments. Subsequently, we will show that training the ensuing DNN on physically extracted moments, rather than true moments, allows to overcome such non-idealities.

V. DISCUSSION AND USE-CASES

A. Analog-Domain Operating Precision Space Exploration

Our analog-domain approach offers several degrees of freedom along which the necessary workload can be minimized. E.g., DAC resolution in the density function synthesizer can be curtailed. Likewise, fewer time steps can be considered to integrate the density functions for moment extraction. Fig. 5(a-c) shows the impact of precision reduction along the above

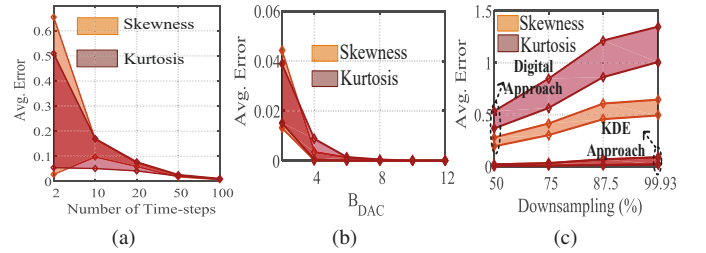


Figure 5: (a) Energy comparison between digital and proposed feature extraction approaches. (b) Impact of number of time-sweep steps and (c) DAC resolution on moment extraction.

time and signal spaces by considering a thousand time-traces from the predictive maintenance dataset [4]. In Fig. 5(a), the best and worst-case error bounds on the extracted features (skewness and kurtosis) *vs.* the number of time-sweep steps are shown where only about a hundred time-steps are needed to extract time-series moments accurately. Similarly, in Fig. 5(b), DAC resolution (B_{DAC}) can be minimized to 6-bit without noticeable accuracy degradation. Many time-series processing approaches also consider signal downsampling to minimize the necessary workload. For example, at $1/N$ downsampling rate, only every N^{th} time series sample can be used for processing. Fig. 5(c) shows the impact of downsampling on the accuracy of extracted features using the proposed approach. The x -axis shows the percentage of samples that can be discarded under downsampling since KDE functions [Eq. (1)] generalize density estimation by signal interpolation; only a limited number of time-traces points are needed to estimate signal density with sufficient accuracy. Therefore, in Fig. 5(c), more than 90% signal samples can be discarded to minimize the workload without incurring significant error considerably.

B. Use-case Study on Predictive Maintenance

We discuss the application for predictive maintenance using NASA's bearing dataset [4] where a deviation from the typical operating characteristics of various machinery is sensed, such as using vibration or sound measurements. On-sensor analytics for predictive maintenance is gaining prominence to minimize the latency [1]. The sensors for predictive maintenance are often placed *ad hoc*. In constricted spaces, on-sensor analytics must require minimal footprint and low operating power. Our analog-domain approach adheres to both of these requirements. Utilizing KDE-based signal interpolation, considerable downsampling can minimize the necessary processing power

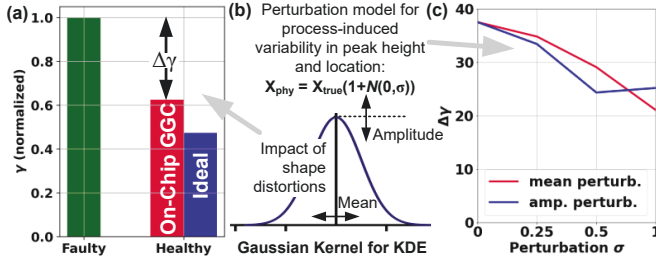


Figure 6: Impact of (a) systematic and (c) process-induced non-idealities in the proposed analog approach. (b) Process variability is emulated by randomly perturbing kernel function mean and amplitude using a Gaussian function.

[Figure 5]. To extract various order moments, our analog-domain processing can reuse the same processing steps and components, such as density estimation in Eq. (2).

An auto-encoder (AE) operates on the feature map extracted by the analog front-end for the studied application. AE is trained on normal time-series data where the encoder compresses the input features to a latent representation which is then reconstructed using the subsequent decoder. Since AE is only trained on normal data, a high reconstruction error arises when AE is fed with anomalous time series. We compute reconstruction error as the average Euclidean distance between the input and output feature map from AE where the input feature map corresponds to statistical moments extracted in our analog-domain approach, i.e., AE is trained based on physically emulated kernel functions instead of true Gaussian functions. An error metric γ is defined by normalizing the reconstruction error on the test data against training data, i.e., $\gamma = \frac{\epsilon_{\text{test}}}{\epsilon_{\text{train}}}$. Since healthy (normal) data can be better reconstructed, its γ is smaller; meanwhile, faulty (anomalous) data has higher γ . Higher discrepancy of γ between healthy and faulty data better identifies predictive maintenance scenarios.

Fig. 6(a) shows the comparison of γ on healthy and faulty data when estimating based on ideal Gaussian functions and based on our on-chip GGCs. For better visualization, γ is normalized against faulty data in both cases. Even though our on-chip kernel functions deviate from ideal Gaussian, by training the subsequent AE on physically implemented kernel function, the proposed analog approach compares well to an ideal implementation. Note that physically implemented kernel function shows many non-idealities such as function amplitude dependence to function mean and asymmetry as seen in Figure 3(b). Fig. 6(c) shows the impact of process variability by randomly perturbing kernel function's mean and amplitude from the designed using perturbation model shown in Fig. 6(b). Analog approach is quite resilient to such perturbations which is attributed to a distributed analytics over many GGCs to minimize the impact of non-idealities of an individual cell.

C. Analog vs. Digital Processing

Unlike in the analog domain, computing with Gaussian kernels in the digital domain is complex, requiring operations such as log-ADD [5] and Gaussian function look-up [3], [6]. Therefore, for digital, we consider moment extraction in discrete domain as $E[x^K] \approx \sum_{i=1}^N x_i^K$. Fig. 4 shows a comparable datapath. Fig. 7 shows the distribution of energy in the

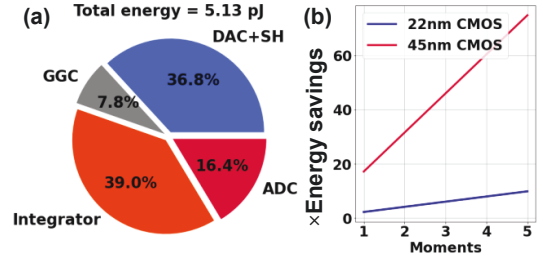


Figure 7: (a) Energy distribution in the proposed analog approach to process a hundred sample time-series. (b) Projected energy savings compared to digital approach (45nm & 22nm) for various order moments.

analog approach for various operations and projected energy savings (in $\times$) over digital implementation using component-level estimates in [7]. For the analog, we consider a signal downsampling of $10\times$. For the discrete processing in digital, the entire trace is considered since it is less amenable to downsampling in Fig. 5(c). The Analog approach's energy dissipation is mostly dominated by the bias power of components such as GGC and OP-AMP; therefore, various order moments require almost similar energy. Comparatively, the workload of the digital approach increases with moment order due to more multiplications per sample. Fig. 7(b) considers 4-bit digital processing with 45nm and 22nm technology. By exploiting signal downsampling and obviating operations such as additions, the proposed analog approach (in 65nm) requires $17\times$ less energy than 45nm digital for the first-order moment and $75\times$ less energy for the fifth-order moment. Although a digital implementation is more technology scalable, the energy advantages of analog approach are still significant when comparing to 22nm projections.

VI. CONCLUSION

We have presented a nonparametric analog-domain feature extraction module to leverage continuous-time signal integration and interpolation to minimize the necessary workload. While analog preprocessing is susceptible to non-idealities, that downstream processing can be co-designed against them.

REFERENCES

- [1] B. et al., "Adepos: A novel approximate computing framework for anomaly detection systems and its implementation in 65-nm cmos," *IEEE Trans. on Circuits and Systems I: Regular Papers*, vol. 67, no. 3, 2020.
- [2] Z. Peng and F. Chu, "Application of the wavelet transform in machine condition monitoring and fault diagnostics: a review with bibliography," *Mechanical Systems and Signal Processing*, vol. 18, no. 2, 2004.
- [3] A. Shylendra, P. Shukla, S. Mukhopadhyay, S. Bhunia, and A. R. Trivedi, "Low power unsupervised anomaly detection by nonparametric modeling of sensor statistics," *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, vol. 28, no. 8, pp. 1833–1843, 2020.
- [4] "NASA bearing dataset," Accessed Jul. 2021. Available: <https://ti.arc.nasa.gov/tech/dash/groups/pcoe/prognostic-data-repository/>.
- [5] P. Shukla, A. Shylendra, T. Tulabandhula, and A. R. Trivedi, "Mc2ram: Markov chain monte carlo sampling in sram for fast bayesian inference," in *International Symposium on Circuits and Systems (ISCAS)*, 2020.
- [6] N. Iliev, A. Gianelli, and A. R. Trivedi, "Low power speaker identification by integrated clustering and gaussian mixture model scoring," *IEEE Embedded Systems Letters*, vol. 12, no. 1, pp. 9–12, 2019.
- [7] X. et al., "Performance comparisons between 7-nm finfet and conventional bulk cmos standard cell libraries," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 62, no. 8, pp. 761–765, 2015.