2023 IEEE 3rd International Conference on Industrial Electronics for Sustainable Energy Systems (IESES) | 979-8-3503-2475-4/23/$31.00 ©2023 IEEE | DOI: 10.1109/IESES53571.2023.10253705

# A Self-powered Sensing System With Embedded TinyML for Anomaly Detection

Zijie Chen, Yiming Gao, and Junrui Liang
School of Information Science and Technology,
ShanghaiTech University, Shanghai 201210, China
Email: {chenzj1, gaoym, liangjr}@shanghaitech.edu.cn

***Abstract*—In the coming Industry 4.0 era, IoT (Internet of Things) technology plays a more and more critical role. Anomaly detection is one of the essential applications for ensuring safe manufacturing, living, etc. It equips the machine with intelligence for perceiving its situation in real-time to prevent disastrous breakdown. Such monitoring systems were usually based on commercial inertial measurement units (IMU) and artificial intelligence (AI) algorithms that run on resource-rich, power-hungry servers, causing a certain amount of energy for sensing and data transmitting. In this paper, we introduce a novel self-powered sensing system with tiny machine learning (TinyML) technique for anomaly detection. A lightweight piezoelectric self-powered sensor (SPS) is utilized to substitute the IMU. The system runs on a low-cost embedded system, realizing the low-power in-situ inferring. A rich dataset has been collected on a vibration platform and analyzed by six well-known machine-learning models. A compressed deep neuron network (DNN) with three hidden layers achieves an accuracy of 97.6% given only 8-point normalized SPS data. The TinyML model is then deployed on embedded systems for on-device inferring and condition-based monitoring. Power measurement is conducted to compare the systems based on an IMU and an SPS. It has shown that the proposed self-powered sensing approach can save up to 66.74% of energy. The system provides a valuable reference for realizing pervasive sensing and ubiquitous AI.**

***Index Terms*—Self-powered sensing, tiny machine learning, anomaly detection, AIoT.**

## I. INTRODUCTION

Modern technologies are developing toward a big picture of Industry 4.0. Ubiquitous connectivity and seamless data exchange are realized among people, machines, and products through the internet [1]–[3]. In this era, efficient maintenance of machines becomes essential due to the high costs resulting from unpredictable machine downtime and defective products [4], [5]. Anomaly detection is a technique utilized for identifying malfunctions in factory machines. It significantly contributes to building self-monitored manufacturing systems [6]–[8]. Machines that are able to perceive possible anomalies can trigger a corresponding maintenance call, thus avoiding the unexpected breakdown of production lines. Recently, machine learning (ML) methods have shown exemplary performance in anomaly detection, and prediction [9]–[11]. The identification accuracy even gets better by using the improved artificial intelligence (AI) techniques [12]. However, the ML approaches mainly rely on resource-rich personal computers (PC) or servers. All raw data sensed by the edge devices are transmitted to the cloud server for analysis, which leads to considerable energy consumption and risk of data privacy leakage [13].

Tiny machine learning (TinyML) is a branch of state-of-the-art ML techniques booming recently. It makes the low-cost edge devices capable of running ML models locally at a milliwatt-level power consumption without real-time support of large servers [14], [15]. The key idea of TinyML is to compress the neural network models, simplify their computational complexity, and make them more power-efficient than conventional computationally expensive and power-hungry ML systems [15], [16]. In addition, the inference at the edge devices reduces potential data security problems.

Many vibration signal analyzers are realized using an IMU (Inertial Measurement Unit) sensor. Yet, IMUs are usually more power-hungry than low-power microcontrollers (MCUs) for supporting their internal conditioning circuit. Its power requirement limits the further development of long-term, and large-scale sensing [17], [18]. To further pursue the ultra-low-power characteristic, many energy harvesting (EH) technologies were proposed with the purpose of transforming the otherwise wasted ambient energy into useful electricity [19]. Before making the whole system self-powered, which is a more demanding task, many studies use the kinetic energy harvesting (KEH) module as a self-powered sensor (SPS) [20]–[22]. In those designs, the parametric relation between the generated voltage and the specific motions is utilized to extract the motion information. Compared with the IMU-based methods, these SPS-based systems can save the power of the sensing part. Distortion caused by the analysis of SPS signals obtained directly at low sampling rates can be compensated by AI techniques.

In this study, we propose a low-cost self-powered sensing system based on TinyML for anomaly detection. It realizes ultra-low-power operation and intelligence with an embedded end device. The purpose is to explore the opportunity of pervasive sensing and ubiquitous AI.

## II. SYSTEM OVERVIEW

Fig. 1 shows the architecture of the proposed novel anomaly detection system, which achieves ultra-low-power and low-cost characteristics. A lightweight piezoelectric cantilever as an SPS is utilized to substitute commercial IMU and form an end-side device with a low-cost MCU. The end device is installed on the monitored equipment for real-time monitoring.

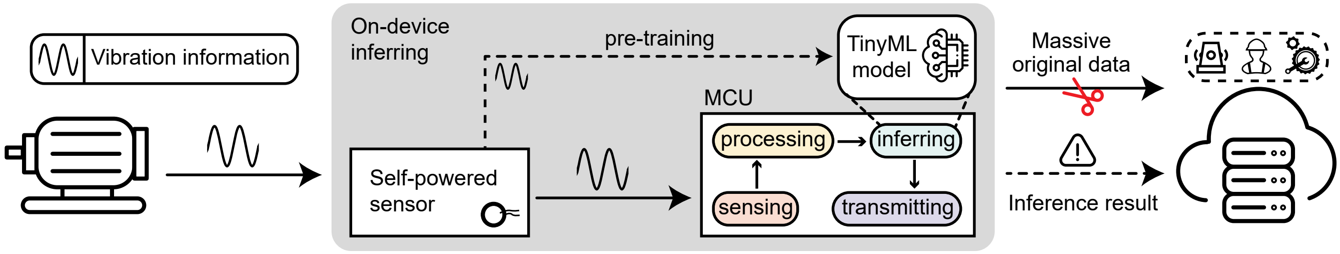


Fig. 1. Architecture of the proposed low-power low-cost on-device anomaly detection system based on TinyML and self-powered sensor.

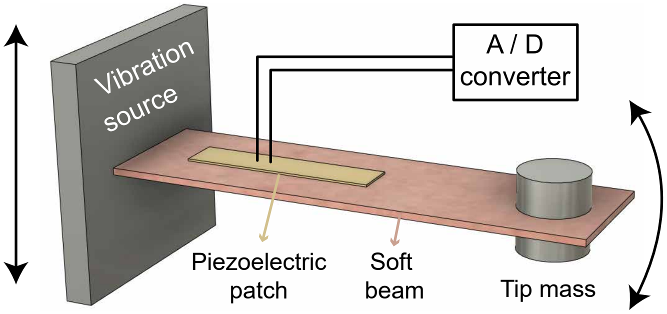


Fig. 2. Base-excited piezoelectric cantilever as a sensor.

A voltage signal is generated from the based-excited self-powered piezoelectric sensor. First, the real-world data for different machine statuses are collected with an ultra-low sampling rate for saving energy and used for pre-training a TinyML model. With the help of experts, the system can identify the occurrence of the anomaly as well as some specific abnormal situations. The end device executes intelligent evaluation when a compressed AI model is deployed. The immediate real-world data is sensed and preprocessed. After on-device inferring, the current machine state is identified and reported to the remote receiver. The MCU can be designed to send an alarm only when a suspicious condition is detected. The local anomaly detection system reduces the continuous data transmission effort, which causes most of the power consumption in the previous designs. The system realizes self-powered vibration sensing for real-time edge machine diagnosis with high accuracy, short inference delay, and minimal development cost, further pushing the boundary of end-point intelligence.

## III. WORKING PRINCIPLE

In the old days, an MCU could not analyze the local raw data due to its limited computing resources and memory. To fulfill the end-side intelligence with ultra-low power consumption, the proposed self-powered vibration sensing system uses a piezoelectric energy harvester as a data source and TinyML as a data inferring method. It operates intermittently to make economical use of energy.

### A. *Self-powered Sensing*

The common IMU-based method senses the vibration using a three-axis accelerometer and transmits the digital data through an $I^2C$ bus, both requiring a stable power supply. A self-powered sensing approach can generate one-axial vibration data without an extra power supply. The power consumption for generating vibration signals, therefore, can be eliminated. A piezoelectric transducer can be considered an SPS requiring no extra power supply. Fig. 2 shows a piezoelectric cantilever that generates a voltage according to the vibration condition. One end of the cantilevered beam is fixed on the vibrating structure, which is usually referred to as the base. The other end is free and mounted with a tip mass. As the machine vibrates, the beam is simultaneously excited by the vibrating base. At the open-circuit condition, the piezoelectric patch subsequently generates a voltage $v_p(t)$, which is proportional to the beam deflection $x(t)$ with a ratio of $\alpha$, i.e.,

$$v_p(t) = \alpha x(t). \tag{1}$$

In this study, the SPS directly connects to the analog-to-digital converter (ADC) of the MCU for data sensing.

However, a simple piezoelectric sensor is less accurate than a well-calibrated integrated IMU. Fig. 3(a) shows the voltage signal generated by a piezoelectric SPS under a 10.972 Hz vibration environment. The sensed value has some degree of noise due to the factors such as nonlinearity, electromagnetic interference, and mechanical resonance. Since the ADC of most MCUs can only read positive voltage, an additional voltage offset circuit is needed to read the negative value. Extra power consumption must be caused by the voltage offset circuit. For the highly low-cost and low-power purpose, the negative part is directly dropped in this study. The digital values in 2 seconds are shown in Fig. 3(b), where the piezoelectric voltage is sampled at 20 Hz with the MCU. Large distortions are caused by discarding the negative values and sampling at such an ultra-low sampling rate. The less required sensing data, the better for energy saving. Fig. 3(c) shows the single-side fast Fourier transform (FFT) values of two randomly picked 16 sequential ADC data points. We found FFT method can well characterize the vibration pattern, even at such a low sampling rate and different sampling phases. However, when it comes to an 8-point situation, it is challenging to draw a simple pattern, as illustrated in Fig. 3(d). While this plot is for one of the vibrational states, plots for other low-frequency situations exhibit a similar characteristic. So it can be expected that an expert system building for the small SPS data size will cost a certain amount of effort.

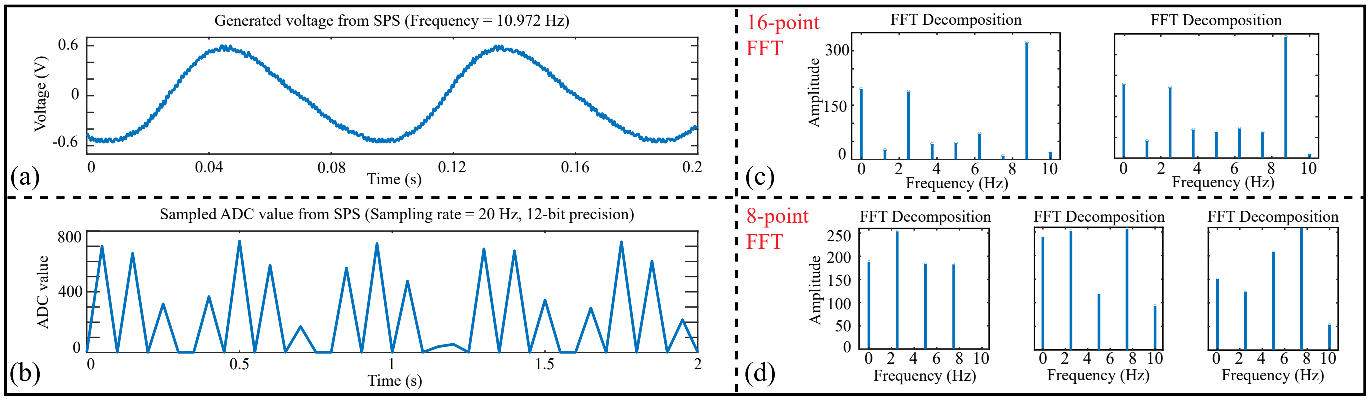


Fig. 3. Data illustration and analysis. (a) Voltage data from a self-powered sensor (SPS) under a 10.972 Hz vibration environment. (b) Digitized values after analog-to-digital conversion at 20 Hz sampling rate from the MCU with an SPS fixed. (c) Two randomly selected sets of single-side FFT values of 16 sequential ADC data points. (d) Three randomly selected sets of single-side FFT values of 8 sequential ADC data points.

Although this distorted signal is hard to analyze using a simple algorithm, we can still use the efficient TinyML technology for realizing the on-device inferring.

### *B. TinyML Development and Deployment*

As mentioned above, TinyML equips the industrial end-machines with basic intelligence. There are three steps for TinyML deployment – end device sampling, model generation, and application development, as depicted in Fig. 4. A self-powered sensor is used to capture the vibration information from the equipment. To further reduce the power consumption, the lower the sampling rate of MCU, the better. Subsequently, the real-world data goes through some preprocessing methods, such as data normalization, missing value imputation, and feature selection [23]. To better achieve high accuracy in identification, a deep neural network (DNN) is built with TensorFlow (Google Inc.) and trained with the processed data. It is worth noting that the smaller the size of the model, the less inference time [24]. After training, the model is compressed by TensorFlow Lite, a cutting-edge inference frame designed for running machine learning algorithms on resource-constrained end devices [25], with skills like parameter quantization and pruning [26]. The fewer data and smaller training epochs required, the lower the development cost. Finally, after being converted into a document suitable for embedded systems, the lightweight model is easily deployed to the TinyML-supported MCUs. Different applications can be easily customized thanks to the complete and open embedded ecosystem.

### *C. On-device Monitoring*

Before locally inferring the real-time signals, the MCU is required to extract some features of the data in time- and frequency-domain for improving the detection performance [18]. FFT is usually carried out with the vibration time-series information, which will definitely cost specific electric energy. In this study, we prove that the raw data is only needed to be normalized before being sent to the TinyML model. The conceptual current profile of the proposed on-device low-power monitoring system is shown in Fig. 5. The system can be based on most 32-bit Cortex-M or Xtensa LX6 MCUs. The whole process can be divided into four types of states: sleeping, sampling, data preprocessing, and inferring. During the normal working process, first, the real-world data is sensed within a specific time interval at a low sampling rate. Then, the original data is processed and inferred on the device in an extremely short time, typically within one second. Studies have already demonstrated that intermittent operation has a significant impact on reducing power consumption [19]. Therefore, the MCU is mostly in sleep mode to save energy. It is awakened by a timer and actively works intermittently. The current is at $\mu$A and mA levels at sleeping and active modes, respectively. MCU has different sleeping modes, such as deep sleep and light sleep. Since the MCU is required to be initialized after waking up from deep sleep, light sleep mode with a relatively higher standby current during sleeping is preferred when the energy cost of MCU initialization is high. In this study, we choose light sleep mode considering the MCU (ESP32, Espressif Inc.) we use. The energy consumption of different MCU operations is discussed in detail in the next chapter.

After the on-device inference, the MCU would transmit a warning message through the Bluetooth low-energy (BLE) or low-power Wifi channel once any anomaly is detected. The power consumption can further be reduced by using the ultra-low-power system on chip (SoC) or energy-aware programming technology [21].

## IV. EXPERIMENTAL EVALUATION

To comprehensively evaluate the performance of the proposed system, we have conducted several experiments. A rich data set is built and analyzed with different AI methods in lab tests. Measurements of energy consumption are also carried out to quantify the performance of power saving.

### *A. Data acquisition*

Maintenance issues can be completely different in many application scenarios, and the anomaly information is particular in each problem [27]. In addition, there is no publicly available

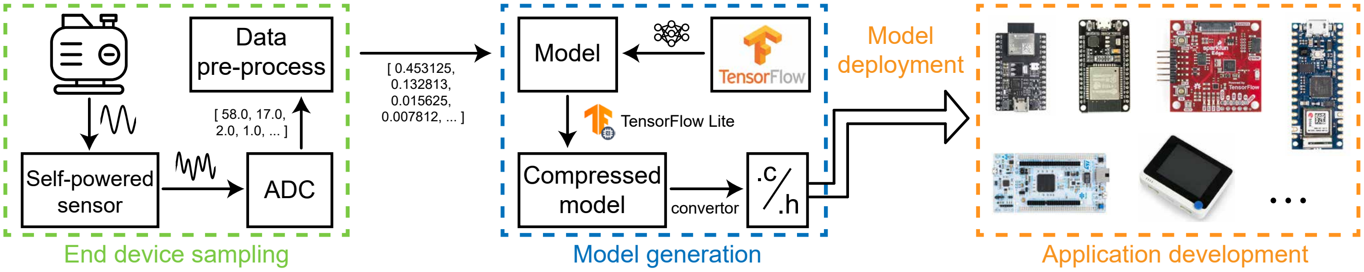


Fig. 4. The development and deployment process of the proposed TinyML-based system toward lightweight demand.

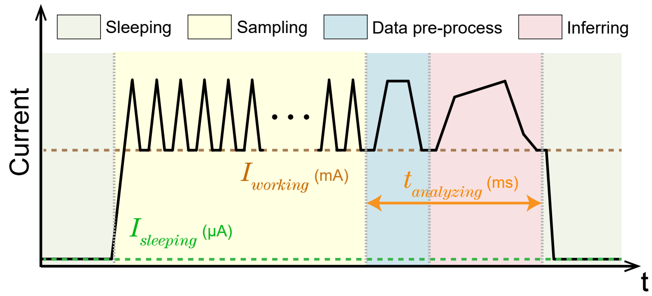


Fig. 5. Current profile illustrating the intermittent operation of the on-device low-power monitoring.

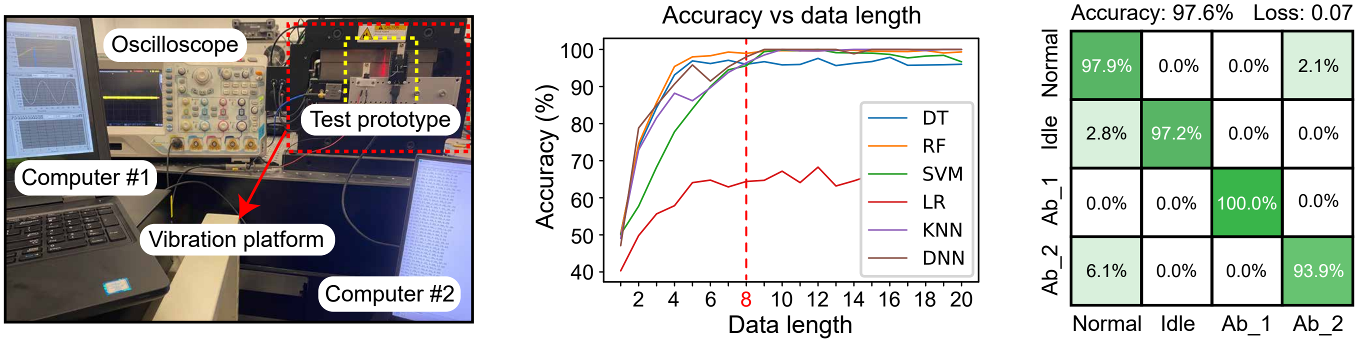
SPS-based dataset for simulation experiments. In this study, we use a vibration platform to simulate a working machine and collect multiple data sets by referring to a previous study [28]. The experimental setup for data acquisition is shown in Fig. 6(a). Two low-cost cantilevered piezoelectric sensors are installed on the vibration source, controlled by a computer-based vibration controller (computer #1). As the platform oscillates, an oscilloscope measures the voltage of one generator. The voltage from the other sensor is sampled by an MCU (ESP32, Espressif Inc.) at a rate of 20 Hz with 12-bit precision. The digital values are sent to another computer (computer #2). A pure 8 Hz vibration is regarded as a normal state, while a 0 Hz one is considered idle. Two unwanted voltage components at 10.972 Hz and 14.656 Hz are considered two specific abnormal conditions. Two unwanted voltage components at 10.972 Hz and 14.656 Hz are regarded as two specific abnormal conditions regarding [28]. Since the variability of the sampling phases in practical applications, we have performed multiple time-staggered data sampling for each vibration case. A dataset of more than 1000 seconds of vibration data against different frequencies has been collected and used for data analysis. The SPS can be regarded as a band-pass filter, mitigating some electromagnetic interference. All vibration cases have been guaranteed to have the same amplitude generated by the SPS, i.e., about 0.6 V, avoiding the machine learning algorithm to judge the vibration cases based on the amplitude. In this first proposal of the anomaly detection design based on TinyML and SPS, we focus on analyzing the more fundamental and predominant periodic vibrations. The non-periodic vibrations or data noises caused by factors such as temperature will be discussed in our future work.

### *B. Machine Learning Analysis*

Given the challenges of the distorted SPS signal, we intend to use six well-known supervised machine learning classification algorithms, including Decision Tree (DT), Random Forest (RF), Support Vector Machine (SVM), Logistic Regression (LR), K-Nearest Neighbor (KNN), and Deep Neural Network (DNN), to analyze the data. For comparability, the analysis is based . Traditional ML models are based on the default settings of the sklearn library, and the results are based on 5-fold cross validation. DNN is set to 2 hidden layers, and the neuron distribution is (16, 8); the activation function is ReLU; the output function is Softmax; the optimizer is adam; the loss function is cross entropy function; the batch size is 16; the learning rate is 0.1. All the crude SPS data is simply normalized before training. Fig. 6(b) shows the accuracy of different models for data length ranging from 1 to 20. Except for the LR algorithm, all models reach a high accuracy when the data length is only 8, which means the window size is 0.4 seconds at a 20 Hz sampling rate. DNN and RF achieve accuracy up to 98.6%. Therefore, AI methods can satisfy high performance in identifying SPS signals with ultra-small window sizes. Following the proposed TinyML pipeline shown in Fig. 4, we choose DNN for TinyML deployment and energy measurement. The comparison of power consumption, deployment cost, latency, etc., for other ML methods will be discussed in our future work. Fig. 6(c) illustrates the confusion matrix of a compressed DNN model for quadruple classification case (0, 8, 10.972, 14.656 Hz), which proves the performance of the proposed system. The hidden layer structure of the DNN is (16, 8, 4), and the size of the compressed model is only 18.8 kilobytes. Since the model has an ultra-high recognition ability for different machine states, when unknown anomalies occur, the MCU will have a relatively low confidence rate, which means the MCU can also accurately discern the occurrence of unknown anomalies with a simple threshold setting.

### *C. Energy Measurement*

Fig. 7 shows the experimental setup for measuring the energy consumption in either the SPS-based or IMU-based system. Two MCUs (ESP32, based on a 240 MHz Tensilica



| | Normal | Idle | Ab_1 | Ab_2 |
|---|---|---|---|---|
| Normal | 97.9% | 0.0% | 0.0% | 2.1% |
| Idle | 2.8% | 97.2% | 0.0% | 0.0% |
| Ab_1 | 0.0% | 0.0% | 100.0% | 0.0% |
| Ab_2 | 6.1% | 0.0% | 0.0% | 93.9% |

(a) (b) (c)

Fig. 6. Setup and results of lab tests. (a) Experimental setup for data acquisition. (b) Model accuracy of different models for data length ranging from 1 to 20. (c) Confusion matrix of a DNN for quadruple classification cases (window size = 0.4 sec).

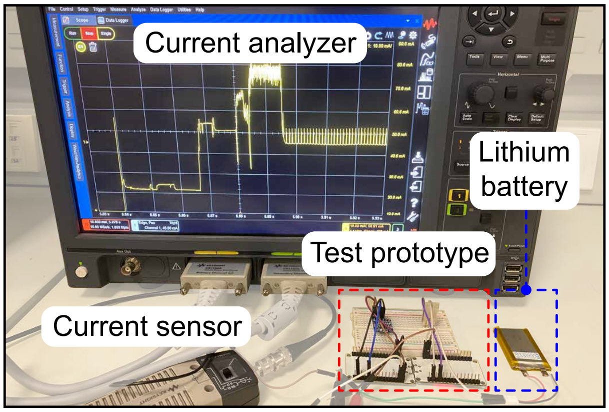


Fig. 7. Experimental setup for energy measurement.

TABLE I
MEASURED ENERGY CONSUMPTION.

| Atomic operation | Processed data | Duration | Energy |
|---|---|---|---|
| SPS & ADC sampling | int16*1 | 0.12 ms | 20.02 $\mu$J |
| IMU sampling | int16*1 | 1.0 ms | 333.50 $\mu$J |
| Normalization | int16*8 | 0.1 ms | 16.64 $\mu$J |
| Inferring | float16*8 | 0.2 ms | 36.71 $\mu$J |
| FFT | int16*8 | 0.3 ms | 56.15 $\mu$J |

TABLE II
COMPARISON OF THE ENERGY CONSUMPTION DURING ONE ROUND OPERATION BETWEEN IMU-BASED AND SPS-BASED SYSTEMS.

| System | Data sampling | Pre-processing | Inferring | Total |
|---|---|---|---|---|
| IMU-based | 3704 $\mu$J | 16.64 $\mu$J | 36.71 $\mu$J | **3757.35 $\mu$J** |
| SPS-based | 1196.16 $\mu$J | 16.64 $\mu$J | 36.71 $\mu$J | **1249.51 $\mu$J** |

Xtensa LX6 processor with 512 KB RAM, 2 MB flash, Espressif Inc.) of the same type are chosen for group experiments. A lightweight off-the-shelf PVDF piezoelectric-film sensor is utilized as an SPS. A commonly used IMU, MPU6050, is used in the comparative study. The IMU is sampled as 100 Hz using the I2C bus embedded in the MCU. For a fair comparison, the IMU is initialized to only sample acceleration in a single direction, generating only 1-axial data. In addition, no third-party library is used for programming, which may add extra power consumption. The MCU is connected with a supporting current sensor in series and powered by a 3.7 V lithium battery. Then, the current is measured by a CX3322A current analyzer (Keysight Inc.) connected across the current sensor.

The energy consumption statistics of a round of SPS & ADC sampling, IMU sampling, data normalization, model inferring, and FFT are listed in Table I. The duration and required energy for model inferring are very low. It proves the rapid and economic characteristics of TinyML and ubiquitous AI. We can see the duration and costed energy of FFT is bigger than that of the TinyML inferring, which means TinyML outperforms FFT in performance and power consumption in this study. In real-world applications, the MCU would go to the light sleep mode before the next sampling operation, and the current from the MCU (ESP32, Espressif Inc.) during light sleep proves 0.8 mA on average. Thus the energy for sleeping during eight-time sampling in 20 Hz sampling rate is $800 \times 3.7 \times 7 \times 0.05 = 1036$ $\mu$J. Thus the energy consumption for data sampling are $333.5 \times 8 + 1036 = 3704 \mu J$, and $20.02 \times 8 + 1036 = 1196.16 \mu J$ for IMU-based and SPS-based systems, respectively. The total energy consumption of the system based on light sleep can be calculated as the sum of the energy for data sampling, pre-processing, and inferring. As referred to Table I, for the IMU-based case, the total consumed energy is $3704 + 16.64 + 36.71 = 3757.35$ $\mu$J, while that of the SPS-based case is $1196.16 + 16.64 + 36.71 = 1249.51$ $\mu$J, as listed in Table II. It demonstrates that the proposed system can save 66.74% of energy for carrying out the same AI-based detection task. The results indicate that SPS is superior to most commonly used accelerometers in terms of energy saving. And the TinyML technique can significantly compensate for the directly obtained distorted data with nearly no energy or latency cost. Note that the self-powered sensing system proves more energy-efficient when using other more low-power MCUs.

The proposed self-powered sensing system with embedded TinyML is general. It can be applied to detecting irregularities and preventing severe events in other application scenarios with vibration signals, such as aero engines, industrial robots, bridges, etc. The on-device, energy-efficient, and low-cost characteristics of the system provide a promising prospect for pervasive sensing and AIoT.

## V. CONCLUSION

This paper proposed a novel low-cost self-powered sensing system with embedded TinyML for anomaly detection application. A piezoelectric SPS was utilized to replace the commonly used IMU. The TinyML technology enabled the end devices to analyze original data locally with high accuracy and low power consumption. This approach cuts off the tight dependency on high-end servers and unleashes a plethora of opportunities for edge intelligence. Toward extreme low-power demand, a low sampling rate of 20 Hz is chosen. A rich dataset based on the SPS was collected with a vibration platform and analyzed by six well-known ML models, including LR, SVM, DT, KNN, RF, and DNN. A DNN with three hidden layers achieved accuracy up to 98.6% and 97.6% for triple and quadruple vibration statuses classification given only 8-point original normalized SPS data. In addition, the MCU works intermittently to extend the battery lifetime economically. After a thorough low-power design from sensing, analyzing, intermittent operation mode, etc., the proposed system can reliably fulfill real-time monitoring with self-powered sensing capability. Several experiments were carried out to validate the system and further quantify its performance and energy consumption. The results showed that the self-powered sensing system achieves excellent performance in anomaly identification while saving 66.74% of energy compared with its IMU counterpart. Such a design provides valuable inspiration for future pervasive sensing and ubiquitous AI studies.

## REFERENCES

[1] A. G. Frank, L. S. Dalenogare, and N. F. Ayala, "Industry 4.0 technologies: Implementation patterns in manufacturing companies," *International Journal of Production Economics*, vol. 210, pp. 15–26, 2019.

[2] T. Borgi, A. Hidri, B. Neef, and M. S. Naceur, "Data analytics for predictive maintenance of industrial robots," in *2017 International Conference on Advanced Systems and Electric Technologies*, 2017, pp. 412–417.

[3] S. Wang, J. Wan, D. Zhang, D. Li, and C. Zhang, "Towards smart factory for industry 4.0: a self-organized multi-agent system with big data based feedback and coordination," *Computer Networks*, vol. 101, pp. 158–168, 2016.

[4] R. K. Mobley, *An Introduction to Predictive Maintenance*. Elsevier, 2002.

[5] M. Haarman, M. Mulders, and C. Vassiliadis, "Predictive maintenance 4.0: predict the unpredictable," *PwC and Mainnovation*, vol. 4, 2017.

[6] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM computing surveys (CSUR)*, vol. 41, no. 3, pp. 1–58, 2009.

[7] A. Kumar, R. B. Chinnam, and F. Tseng, "An hmm and polynomial regression based approach for remaining useful life and health state estimation of cutting tools," *Computers & Industrial Engineering*, vol. 128, pp. 1008–1014, 2019.

[8] A. Turnbull, J. Carroll, and A. McDonald, "Combining scada and vibration data into a single anomaly detection model to predict wind turbine component failure," *Wind Energy*, vol. 24, no. 3, pp. 197–211, 2021.

[9] T. P. Carvalho, F. A. Soares, R. Vita, R. d. P. Francisco, J. P. Basto, and S. G. Alcalá, "A systematic literature review of machine learning methods applied to predictive maintenance," *Computers & Industrial Engineering*, vol. 137, p. 106024, 2019.

[10] Q. Cao, C. Zanni-Merk, A. Samet, C. Reich, F. d. B. de Beuvron, A. Beckmann, and C. Giannetti, "Kspmi: a knowledge-based system for predictive maintenance in industry 4.0," *Robotics and Computer-Integrated Manufacturing*, vol. 74, p. 102281, 2022.

[11] M. Baptista, S. Sankararaman, I. P. de Medeiros, C. Nascimento Jr, H. Prendinger, and E. M. Henriques, "Forecasting fault events for predictive maintenance using data-driven techniques and arma modeling," *Computers & Industrial Engineering*, vol. 115, pp. 41–53, 2018.

[12] K. Kaur, M. Selway, G. Grossmann, M. Stumptner, and A. Johnston, "Towards an open-standards based framework for achieving condition-based predictive maintenance," in *Proceedings of the 8th International Conference on the Internet of Things*, 2018, pp. 1–8.

[13] M. Monfort Grau, "Tinyml: From basic to advanced applications," B.S. thesis, Universitat Politècnica de Catalunya, 2021.

[14] R. Sanchez-Iborra and A. F. Skarmeta, "Tinyml-enabled frugal smart objects: Challenges and opportunities," *IEEE Circuits and Systems Magazine*, vol. 20, no. 3, pp. 4–18, 2020.

[15] V. J. Reddi, B. Plancher, S. Kennedy, L. Moroney, P. Warden, A. Agarwal, C. Banbury, M. Banzi, M. Bennett, B. Brown *et al.*, "Widening access to applied machine learning with tinyml," *arXiv preprint arXiv:2106.04008*, 2021.

[16] C. Banbury, C. Zhou, I. Fedorov, R. Matas, U. Thakker, D. Gope, V. Janapa Reddi, M. Mattina, and P. Whatmough, "Micronets: Neural network architectures for deploying tinyml applications on commodity microcontrollers," *Proceedings of Machine Learning and Systems*, vol. 3, pp. 517–532, 2021.

[17] S. Khalifa, M. Hassan, and A. Seneviratne, "Pervasive self-powered human activity recognition without the accelerometer," in *2015 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, 2015, pp. 79–86.

[18] S. Khalifa, G. Lan, M. Hassan, A. Seneviratne, and S. K. Das, "Harke: Human activity recognition from kinetic energy harvesting data in wearable devices," *IEEE Transactions on Mobile Computing*, vol. 17, no. 6, pp. 1353–1368, 2017.

[19] Z. Chen, F. Gao, and J. Liang, "Kinetic energy harvesting based sensing and iot systems: a review," *Frontiers in Electronics*, vol. 3, p. 1017511, 2022.

[20] J. Zhao and Z. You, "A shoe-embedded piezoelectric energy harvester for wearable sensors," *Sensors*, vol. 14, no. 7, pp. 12 497–12 510, 2014.

[21] X. Li, H. Tang, G. Hu, B. Zhao, and J. Liang, "ViPSN-pluck: A transient-motion-powered motion detector," *IEEE Internet of Things Journal*, vol. 9, no. 5, pp. 3372–3382, 2021.

[22] W. Xu, G. Lan, Q. Lin, S. Khalifa, M. Hassan, N. Bergmann, and W. Hu, "KEH-gait: Using kinetic energy harvesting for gait-based user authentication systems," *IEEE Transactions on Mobile Computing*, vol. 18, no. 1, pp. 139–152, 2018.

[23] S. García, S. Ramírez-Gallego, J. Luengo, J. M. Benítez, and F. Herrera, "Big data preprocessing: methods and prospects," *Big Data Analytics*, vol. 1, no. 1, pp. 1–22, 2016.

[24] A. Montanari, M. Sharma, D. Jenkus, M. Alloulah, L. Qendro, and F. Kawsar, "eperceptive: energy reactive embedded intelligence for batteryless sensors," in *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*, 2020, pp. 382–394.

[25] R. David, J. Duke, A. Jain, V. Janapa Reddi, N. Jeffries, J. Li, N. Kreeger, I. Nappier, M. Natraj, T. Wang *et al.*, "Tensorflow lite micro: Embedded machine learning for tinyml systems," *Proceedings of Machine Learning and Systems*, vol. 3, pp. 800–811, 2021.

[26] Y. Zhang, W. Ding, and C. Liu, "Summary of convolutional neural network compression technology," in *2019 IEEE International Conference on Unmanned Systems (ICUS)*, 2019, pp. 480–483.

[27] M. Luo, Z. Xu, H. L. Chan, and M. Alavi, "Online predictive maintenance approach for semiconductor equipment," in *IECON 2013-39th Annual Conference of the IEEE Industrial Electronics Society*, 2013, pp. 3662–3667.

[28] T. Plante, A. Nejadpak, and C. X. Yang, "Vibration analysis: Fault detection and failure prediction," *University of North Dakota, Grand Forks, ND*, 2013.