Practical Implementation of Upgraded Low-Cost Sensors in Everyday Home Devices

Seokho Ahn[a], Hyungjin Kim[a], Euijong Lee[b], Young-Duk Seo[a, †]
Department of Electrical and Computer Engineering, Inha University, South Korea[a],
School of Computer Science, Chungbuk National University, South Korea[b]
sokho0514@gmail.com, flslzk@gmail.com, kongjjagae@cbnu.ac.kr, mysid88@inha.ac.kr

2024 IEEE International Conference on Consumer Electronics (ICCE) | 979-8-3503-2413-6/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICCE59016.2024.10444284

***Abstract*—The crucial part of IoT-controlled devices is the collection of accurate data. However, manufacturers often use low-cost sensors to make everyday home devices affordable, which can compromise accuracy. Therefore, we introduce a novel framework designed to improve the calibration performance of low-cost sensors incorporated into these devices. Applying this framework to home appliances makes it possible to calibrate low-cost sensors with inference speeds comparable to linear models while achieving accuracies similar to those of deep learning models. Specifically, the framework offers a selection of three different model variants, each considering factors such as implementation difficulty, calibration accuracy, or inference speed. Experimental findings indicate that our framework exhibits superior performance in both general-purpose and embedded hardware, highlighting its potential applicability to everyday home devices such as IoT-controlled appliances.**

***Index Terms*—Internet of Things, deep learning, home device, sensor calibration**

I. Introduction

THE development of the Internet of Things (IoT) has enhanced the capabilities of daily home devices [1]. By integrating sensors and software technologies, IoT-controlled products can now offer a broader range of services. Nonetheless, to make these devices affordable for prospective consumers, manufacturers often incorporate cost-effective sensors that can compromise accuracy. Therefore, researchers have utilized deep learning methods to calibrate sensor data, resulting in remarkable improvements in accuracy for some large-scale industrial environments [2]–[4]. Despite recent advancements, there remains limited research on enhancing the performance of low-cost sensors used in smaller-scale environments such as fine-dust, temperature, and humidity sensors, by applying deep learning.

The lack of those studies can be attributed to the simultaneous requirement to address the trade-off of three inherent challenges when calibrating a low-cost sensor using a deep learning model. Specifically, there's a trade-off between three primary challenges: achieving high calibration accuracy, ensuring fast inference speed, and calibrating within constrained hardware resources. As a result, IoT-controlled products typically rely on linear calibration methods rather than deep learning models [5]–[7].

To fill this necessary research gap, we introduce a novel calibration framework for improving the performance of low-cost sensors for everyday home devices. The framework presents three distinct variants of models, and each of these variants operates by adjusting the utilization frequency of the deep learning calibration model in a different way. This design provides users with the flexibility to select a model based on various considerations such as desired accuracy, hardware resource constraints given IoT-controlled products, or implementation difficulty. Every model offers calibration accuracy comparable to deep learning-based models and maintains inference speeds similar to linear models. Notably, the third model variant demonstrates the ability to mitigate the noise inherent in the calibration results, yielding the most accurate calibration results among existing deep learning models. Experimental results reveal that our framework outperforms the baseline methods in terms of both accuracy and inference speed. We further demonstrate its superior performance on embedded hardware, emphasizing its potential applicability for everyday home devices.

II. Related Work

Initially, basic calibration techniques such as moving averages and mathematical filters like the HP filter and Kalman filter were utilized [5]. These choices were influenced by constraints in hardware resources and technological capabilities at the time. However, these methods do not demonstrate satisfactory performance; rather, it exhibits a close correlation with the analytical techniques employed for extracting trends in time-series data. Consequently, it has primarily been utilized as a data preprocessing procedure in recent years [7].

Since then, various machine learning-based and statistical-based approaches have been proposed to calibrate low-cost sensors [6], [7]. These approaches mainly include linear regression, SVM, random forest, XGBoost, and the ARIMA model [6]–[9]. Among these, linear regression has emerged as a prevalent choice due to its simplicity and efficient performance. Nevertheless, these methods exhibit distinct limitations in terms of calibration accuracy. Specifically, the use of linear regression for calibrating low-cost sensor data often fails to capture the nonlinear trends present in time-series data [6]. Moreover, their applicability and scalability are compromised due to their formulation-based modeling paradigm [7].

Recently, various data-driven calibration methods based on deep learning have been introduced. Notably, calibration models employing Long Short-Term Memory (LSTM) [10],

† Corresponding Author

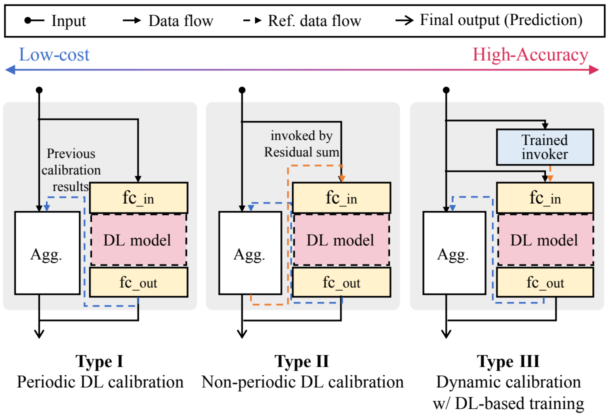

Fig. 1. **Overview of the framework for upgrading low-cost sensors in daily home devices.** One of the three model variants can be selected, considering factors such as implementation complexity, calibration accuracy, or latency.

Convolutional Neural Network (CNN) [11], and Transformer [12] have been extensively explored [6], [7], [13], [14]. One notable advantage of these approaches is their ability to adaptively learn models, potentially leading to high accuracy when sufficient data are available [6]. However, these methods have the disadvantage of high computing resources required for learning [7]. Thus, IoT-controlled products with limited hardware resources face challenges in leveraging complex models [6]. Consequently, despite the recent advancements in accuracy achieved by Transformer, their application in low-cost embedded environments remains limited. Most of them are used by LSTM and CNN-based models, but even those are being used only in limited scenarios [6], [7], [13], [14]. In summary, studies applying existing deep learning models are not suitable for daily home devices, as they overlook hardware resource constraints.

III. Framework for Upgrading Low-Cost Sensors in Everyday Home Devices

In this section, we introduce an efficient deep learning-based framework optimized for the calibration of low-cost sensors in daily home devices. Figure 1 illustrates the whole architecture of this framework. Our framework supports three distinct calibration types, **Type I**, **II**, and **III**. Each of these calibration models is designed to leverage deep learning algorithms without compromising inference speed. The primary difference among these models is the balance they strike between inference speed and model accuracy. For instance, Type I model uses simple equations to make fast inference speed and ease of implementation, however, it does not significantly enhance accuracy. In contrast, Type III model achieves superior accuracy by selectively invoking a deep learning model through parameter training, although its inference speed may be comparatively slower than other types. Detailed explanations of each type are provided in the following subsection.

A. *Type I: Periodic deep learning calibration*

Data collected from sensors in daily life often remains consistent and exhibits change triggered by specific events, such as peaks. Consequently, employing a deep learning model for calibration at all times would not be an efficient approach. Based on this hypothesis, achieving a faster inference speed without compromising accuracy can be attained by simply adjusting the period p of the deep learning model $\mathcal{F}$, i.e.,

$$\hat{y}_i = \begin{cases} \mathcal{F}(\mathcal{S}_i), & \text{if } i \equiv 1 \bmod p \\ \hat{y}_{i-1}, & \text{otherwise} \end{cases} \tag{1}$$

where $\mathcal{S}_i := (x_{i-N+1}, \cdots, x_i)$ denotes i^{th} time-series of sensor measurements x_i's with window size N, and $\hat{y}_i$ is its corresponding i^{th} calibration result. Interpolating inputs can also be employed to calibrate where the deep learning model is not utilized. In this case, however, the previous calibration result is used without an additional method to fully maximize the benefits of the model inference speed.

B. *Type II: Non-periodic deep learning calibration*

The Type I model exhibits rapid inference speed; however, it lacks the capacity to accurately capture the inherent features of a given time-series. For instance, in cases where measurements exhibit frequent changes, it is advisable to employ a deep learning model to reflect these variations. Conversely, when such changes are infrequent, frequent utilizing the deep learning model may become unnecessary. Taking these factors into consideration, the Type II model utilizes a formula that can dynamically adjust the period as the input changes:

$$\hat{y}_i = \begin{cases} \mathcal{F}(\mathcal{S}_i), & \text{if } \dfrac{\mathbf{w} \cdot (\mathcal{S}_i / \mathcal{S}_{i-1} \ominus \mathbf{1})}{\mathbf{w} \cdot \mathbf{1}} > \theta \\ \hat{y}_{i-1}, & \text{otherwise} \end{cases} \tag{2}$$

where $\ominus$ and θ are an element-wise absolute difference operator and a threshold. The weighting factor $\mathbf{w} \in \mathbb{R}^N$ causes the term $\mathbf{w} \cdot (\mathcal{S}_i / \mathcal{S}_{i-1} \ominus \mathbf{1}) / \mathbf{w} \cdot \mathbf{1}$ to compute the weighted moving average of the rate of change for given time-series. Utilizing this equation invokes a deep learning model with a dynamic period based on the given time-series.

C. *Type III: Training-based deep learning calibration*

The Type II model has the ability to dynamically calibrate the deep learning model over varying time periods. However, its structure lacks the capability to learn the temporal patterns of time-series data. In contrast, the Type III model introduces a linear layer $\mathcal{L}$ (referred to as the *invoker*) with the activation function $\sigma_{\mathcal{L}}$ to efficiently capture the potential features. The Type III model is defined as follows:

$$\hat{y}_i = \begin{cases} \mathcal{F}(\mathcal{S}_i), & \text{if } i = 1 \text{ or } \mathcal{L}\left(\mathcal{S}_i \| \left(\hat{y}_{i-1}\right)\right) > \theta_{\mathcal{L}} \\ \hat{y}_{i-1}, & \text{otherwise} \end{cases} \tag{3}$$

where $\theta_{\mathcal{L}}$ is a threshold and $\|$ denotes the vector concatenate operator. The invoker $\mathcal{L}$ is pre-trained by automatically created binary class labels y_i''s which are of the form:

$$y_i' = \begin{cases} 1, & \text{if } i = 1 \text{ or } |\hat{y}_{i-1} - y_i| < |x_i - y_i| \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

where y_i is the i^{th} ground-truth of calibration model. Note that the invoker $\mathcal{L}$ is primarily designed for computational efficiency, resulting in faster inference speeds compared to

TABLE I
MAIN EXPERIMENTAL RESULTS IN TERMS OF RMSE, AVERAGE CPU INFERENCE SPEED (LATENCY).

Model	Method	RMSE↓				Latency↓ (μs)			
		Env1	Env2	Env3	**Avg**	Env1	Env2	Env3	**Avg**
Linear	Vanilla	8.7967	2.9315	9.5209	7.0830	0.1725	0.1642	0.1788	0.1718
LSTM	Vanilla	8.3825	2.3696	8.5640	**6.4387**	0.2887	0.2974	0.3003	0.2955
	Ours (Type I)	8.5094	2.3686	8.5756	6.4845	**0.0278**	**0.0294**	**0.0302**	**0.0291**
	Ours (Type II)	8.5321	2.3376	**8.5589**	6.4762	0.0814	0.1220	0.1589	0.1207
	Ours (Type III)	**8.3737**	**2.3039**	8.6971	6.4582	0.2738	0.1611	0.2522	0.2290
Transformer	Vanilla	8.4652	2.3480	8.5119	6.4417	0.7752	0.7580	0.7671	0.7667
	Ours (Type I)	8.5956	2.3465	8.5220	6.4880	**0.0823**	**0.0781**	**0.0785**	**0.0796**
	Ours (Type II)	8.4768	2.3285	8.5067	6.4373	0.1669	0.2679	0.3303	0.2550
	Ours (Type III)	**8.3820**	**2.2539**	**8.4213**	**6.3524**	0.5336	0.4653	0.4535	0.4841

the deep learning model. Nonetheless, the invoker leverages the time-series $\mathcal{S}_i$ with the past calibration results $\hat{y}_{i-1}$ from the deep learning model, effectively reducing noise within the results. Consequently, the Type III model can achieve greater accuracy compared to only untilizing the deep learning model.

IV. EXPERIMENT

We conducted experimental evaluations of the proposed model in various environments and verify its superiority in terms of model accuracy and inference speed. The baselines for comparative experiments are linear regression, LSTM (Long short-term memory), and Transformer, which are prevalent choices for time-series data. The deep learning baseline models (i.e., LSTM and Transformer) are compared against both the vanilla model (Vanilla) and the proposed model variants (Type I, II, and III).

A. Dataset

Due to the diverse range of sensor types and environments encountered in daily life, there is currently a lack of benchmark datasets suitable for comprehensively evaluating the capabilities of these sensors. Therefore, we performed the task of collecting and refining data using various sensors and environments in our experiment. We selected fine-dust data due to its ubiquity in daily-life sensor usage and its practicality for short-term data collection, essential for training purposes.

There are various types of fine dust sensors, with beta-ray sensors applied in air quality monitoring stations demonstrating the highest accuracy. However, the excessively high cost of beta-ray sensors prevents their practical incorporation into daily devices such as home appliances. Hence, we utilized two types of fine-dust sensors for our experiments: the infrared-type sensor PPD42NS and the laser-type sensor PMS7003. These sensors are commonly utilized in IoT-controlled appliances. Eight PMS7003 sensors and four PPD42NS sensors were employed to measure particulate matter generated in various daily environments, including activities such as turning on/off candles, cooking, laundry, and ventilation.

Data were collected from two distinct types of fine-dust sensors within three distinct real-world environments for 5-10 days each. These environments are labeled as Env1, Env2, and Env3. All data were refined to the 15-second frequency to a 5 minutes window size for the purpose of training. During the refinement process, both a simple moving average (SMA) and Hodrick-Prescott (HP) filters were applied to smoothen the data. Subsequently, the data was synchronized through linear interpolation to ensure measurement at regular time intervals.

B. Experimental Setting

In order to train the deep learning model employed within the experiment, it is essential to gather input sensor data and corresponding ground truth in the same spatial and temporal conditions. Given that laser-type sensors are typically both more accurate and more expensive than infrared sensors, we designated the PMS7003 as a high-cost, high-accuracy sensor, while the PPD42NS as a low-cost and low-accuracy sensor. Additionally, we configured the PPD42NS to measure particulates over PM10 in our experiments. Hence, we set the average PM10 values obtained from the PMS7003 as the ground truth to match the measurements from each low-cost sensor. The model performance in each environment was calculated by averaging the results derived from each PPD42NS.

For evaluating the model performance, we utilize RMSE (Root Mean Square Error) for accuracy assessment and μs (microseconds) for measuring inference speed. Additionally, we implement the proposed model variants in an embedded environment to validate their practicality and applicability.

More implementation details are as follows:

- We implemented all models using PyTorch.
- All models are converted and evaluated to the ONNX format for the purpose in the embedded environment.
- The models were trained on a machine equipped with an Intel i9-13900K 3.00GHz 24-Core processor and an NVIDIA GeForce RTX 3080 graphics card.
- We employed an anchored walk-forward optimization [15] to evaluate the performance of the model using 5-fold cross-validation.
- The window size N was fixed at 20.
- The hidden size of LSTM and Transformer utilized in the experiment was set to 16 and 4, respectively.

C. Evaluation Results

Table I presents the comprehensive performance evaluation result of the proposed framework in our experiment. Our framework shows outstanding performance in terms of RMSE, average CPU latency. Firstly, in terms of model inference

TABLE II
INFERENCE LATENCY RESULTS ON EMBEDDED HARDWARE.

Model	Method	Env1	Env2	Env3	Avg
Linear	Vanilla	0.0769	0.0770	0.0773	0.0771
LSTM	Vanilla	0.1358	0.1358	0.1369	0.1362
	Ours (Type I)	**0.0151**	**0.0151**	**0.0152**	**0.0152**
	Ours (Type II)	0.1357	0.1597	0.1754	0.1569
	Ours (Type III)	0.1232	0.1119	0.1130	0.1160
Transformer	Vanilla	0.2091	0.2092	0.2091	0.2092
	Ours (Type I)	**0.0226**	**0.0226**	**0.0228**	**0.0227**
	Ours (Type II)	0.1574	0.1917	0.2190	0.1894
	Ours (Type III)	0.1733	0.1931	0.2120	0.1928

speed, all model types significantly outperform the baseline deep learning models. Notably, the Type I model exhibits the highest speed, surpassing even the linear model. This superiority can be attributed to its design, which prioritizes maximum inference speed. Secondly, with regard to accuracy, all model types exhibit performance comparable to that of baseline deep learning models, notably surpassing the accuracy of linear models. Particularly noteworthy is the accuracy of the Type III model, which exceeds that of the baseline model. This achievement is attributed to the Type III model's invoker effectively mitigating the noise inherent in the results.

D. Application to everyday home devices

To verify the practical applicability of this study, the proposed model was implemented within embedded environments utilized on IoT-controlled products. Considering the inherent challenges associated with training models in resource-constrained environments such as home appliances, we assume that only weights learned from a server can be deployed in the embedded platforms. Table II presents the comparison results in terms of inference speed, which was measured by integrating the trained model into the IoT home appliance prototype configured as illustrated in Figure 2. As indicated in Table II, similar results were achieved when conducting the same experiment subsequent to converting the trained model variants into the ONNX format. These findings indicate that our proposed model variants are a practical solution for embedded platforms and emphasize their potential for integration into real-world applications.

V. CONCLUSION

This study introduces a novel framework designed to enhance the performance of low-cost sensors integrated into everyday daily devices. The framework offers a selection of three different model variants, each considering factors such as implementation complexity, calibration accuracy, and inference speed. Experimental findings indicate that our framework exhibits superior performance compared to baselines in terms of accuracy and inference speed. We also demonstrate superior performance on embedded hardware, highlighting its potential applicability to everyday home devices.

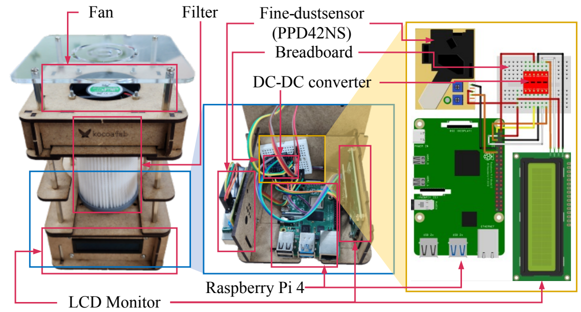

Fig. 2. **Designing a prototype home devices with a proposed framework** and circuit diagram example for Raspberry Pi 4.

ACKNOWLEDGMENTS

This work was supported in part by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (NRF-2022R1C1C1012408), in part by Institute of Information & communications Technology Planning & Evaluation (IITP) grants funded by the Korea government (MSIT) (No.RS-2022-00155915, Artificial Intelligence Convergence Innovation Human Resources Development (Inha University), and No.2022-0-00448, Deep Total Recall: Continual Learning for Human-Like Recall of Artificial Neural Networks, 10%).

REFERENCES

[1] K. K. Patel and S. M. Patel, "Internet of things-iot : Definition , characteristics , architecture , enabling technologies , application & future challenges," 2016.
[2] A. Gupta, A. Murali, D. Gandhi, and L. Pinto, "Robot learning in homes: Improving generalization and reducing dataset bias," 2018.
[3] W. Ke, D. Huang, F. Yang, and Y. Jiang, "Soft sensor development and applications based on lstm in deep neural networks," 11 2017, pp. 1–6.
[4] Q. Sun and Z. Ge, "A survey on deep learning for data-driven soft sensors," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 9, pp. 5853–5866, 2021.
[5] M. D. Taylor, "Calibration and Characterization of Low-Cost Fine Particulate Monitors and their Effect on Individual Empowerment," 12 2016.
[6] F. Concas, J. Mineraud, E. Lagerspetz, S. Varjonen, X. Liu, K. Puolamäki, P. Nurmi, and S. Tarkoma, "Low-cost outdoor air quality monitoring and sensor calibration: A survey and critical analysis," *ACM Trans. Sen. Netw.*, vol. 17, no. 2, may 2021.
[7] P. Nath, P. Saha, A. I. Middya, and S. Roy, "Long-term time-series pollution forecast using statistical and deep learning methods," *Neural Computing and Applications*, vol. 33, no. 19, pp. 12 551–12 570, 2021.
[8] K. Aula, E. Lagerspetz, P. Nurmi, and S. Tarkoma, "Evaluation of low-cost air quality sensor calibration models," *ACM Trans. Sen. Netw.*, vol. 18, no. 4, dec 2022.
[9] E. Villanueva, S. Espezua, G. Castelar, K. Diaz, and E. Ingaroca, "Smart multi-sensor calibration of low-cost particulate matter monitors," *Sensors*, vol. 23, no. 7, p. 3776, Apr 2023.
[10] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 11 1997.
[11] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, F. Pereira, C. Burges, L. Bottou, and K. Weinberger, Eds., vol. 25. Curran Associates, Inc., 2012.
[12] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," 2017.
[13] S. Ali, F. Alam, K. M. Arif, and J. Potgieter, "Low-cost co sensor calibration using one dimensional convolutional neural network," *Sensors*, vol. 23, no. 2, p. 854, Jan 2023.
[14] I. D. Apostolopoulos, G. Fouskas, and S. N. Pandis, "Field calibration of a low-cost air quality monitoring device in an urban background site using machine learning models," *Atmosphere*, vol. 14, no. 2, p. 368, Feb 2023.
[15] P. Nousi, A. Tsantekidis, N. Passalis, A. Ntakaris, J. Kanniainen, A. Tefas, M. Gabbouj, and A. Iosifidis, "Machine learning for forecasting mid-price movements using limit order book data," *Ieee Access*, vol. 7, pp. 64 722–64 736, 2019.