Conceptualising a Benchmarking Platform for Embedded Devices

Asier Garcia-Perez*, Raúl Miñón*, Ana I. Torre-Bastida*,
Josu Diaz-de-Arcaya*, Ekaitz Zulueta-Guerrero‡,
* TECNALIA, Basque Research & Technology Alliance (BRTA)
Parque Tecnológico de Álava Albert Einstein 28, 01510 Vitoria-Gasteiz, Álava, Spain
{asier.garcia, raul.minon, josu.diazdearcaya, isabel.torre}@tecnalia.com
‡ University of the Basque Country (UPV/EHU)
Nieves Cano, 12, 01006 Vitoria-Gasteiz, Álava, Spain
ekaitz.zulueta@ehu.eus

***Abstract*—In the current era, a notable presence of data-generating devices is being witnessed which spans across various sectors like industries, healthcare or smart cities. The diversity of options in the market pose a significant challenge for companies facing the task of selecting the most suitable embedded devices for their specific needs in their particular environments. Several benchmarking solutions have been conducted to overcome this barrier. Nevertheless, generally, they are ad-hoc solutions for a given experiment, which makes it impossible to extrapolate the results to future scenarios. To address this uncertainty and facilitate informed decision-making, this paper presents the conceptualization of a modular and extensible benchmarking platform. This platform has been designed to adapt to the changing dynamics of the market and the specific needs of users. Its modular and extensible approach allows companies to assess and select integrated devices more efficiently, while providing the necessary flexibility to address new and diverse situations.**

***Index Terms*—embedded device, edge computing, flexible benchmarking, modular and extensible**

I. Introduction

Embedded computing is experiencing a reborn due to the proliferation of IoT devices to manage the data generated at the edge layer. As a result, in the last years, several new devices from diverse families are emerging like Field Programmatic Gate Array (FPGA), Micro Controller Unit (MCU) or System on Chip (SoC) [1]. Moreover, embedded artificial intelligence (AI) paradigm is getting more attention from both academy and industry and, consequently, some hardware manufacturers have started to provide their boards with AI accelearators [2]. For instance, Google Coral Dev board is set with a Tensor Processing Unit (TPU) to accelerate the AI management [3]. This situation has significantly boosted the possibility to manage data at the edge layer [4].

Nevertheless, the vast number of emerging devices generates considerable confusion when companies must decide which embedded device to select to satisfy their needs. As a consequence, a benchmarking mechanism offering comparisons among device characteristics, for specific contexts and, in turn, to assist companies in elucidating the best option becomes fundamental.

For this purpose, it is crucial to analyse several key aspects. For instance, application-specific performance, as each task may have particular requirements that need to be evaluated in detail. Additionally, resource limitation plays a key role in edge environments, where benchmarking facilitates the assessment of how efficiently devices can execute their functions. In order to acquire information about these metrics and additional ones like the time taken or the disk, CPU and RAM saturation, several information are necessary, like the experiment to be carried out, the available devices to test and their underlying hardware resources, accelerators and software libraries supported. This leads to the conclusion that the benchmarking platforms must be flexible enough to deal with this set of requirements and also flexible enough to manage additional ones when technologies or specific contexts evolve. Nevertheless, this situation is far from reality since the analysed benchmarking platforms for embedded devices have been conceptualised to conduct ad-hoc experiments to test a specific set of three-four devices with a reduced number of processes or AI models. Therefore, there exists a gap to provide companies mechanisms to easily and rapidly perform specific experiments adapted to their needs without having to create a benchmarking platform from scratch.

Moreover, embedded devices usually do not work alone but are part of specific architectures where data and task can be distributed. This implies a sophisticated utilization of the device where the behaviour can differ to the one when working isolated. Consequently, this would be a more than interesting feature for a benchmarking platform which again has not been identified in the state of the art.

The contribution of this paper consists in conceptualising a modular and extensible benchmarking platform to help companies in the selection of a specific embedded device for given requirement. This platform will be able to deal with new scenarios configured by end users instead of providing only ad-hoc functionality. To this end, a description language will be designed in order to model the relevant key aspects to test the adequacy of such devices like the software and hardware resources, the criteria to consider, the target scenario or the type of task to deploy.

Section II presents the related work. Next, Section III proposed a preliminary approach. Finally, the conclusions an future work are presented in Section IV.

II. Related Work

In this section, various benchmark studies on embedded devices are evaluated. Additionally, Table I presents a summary of the main features considered for the conceptualization of the embedded benchmarking platform contrasted with the evaluated works. Specifically, the features analysed are:

- **Device adaptability:** the ability to conduct experiments on a wide range of devices, covering not only the devices on which the benchmark was initially conceptualised.
- **Custom metrics**: the goal is to achieve an exploration of much more specific and customized metrics that can be defined and selected by the users. Different use cases or user profiles may require additional metrics, and the omission of these could limit the applicability of the results in specific contexts.
- **Artifact diversity:** artifacts, considered as data, models or software processes, should be manageable to enable support for the wider set of testing scenarios. The results obtained may show significant variations when employing different artifacts on the same devices. This limitation underscores the need for a more comprehensive and diversified evaluation of artifacts to enhance our understanding of embedded device performance in real-world usage scenarios.
- **Artifact deployment:** such artifacts should be automatically deployed. In order to reduce the barrier to utilise embedded devices, inexperienced companies should minimise their learning curve and, consequently, they should not initially deal with deployments aspects.
- **Architecture support:** the possibility to benchmark not only a specific device but a device ecosystem where specific devices behaviour could potentially be modified.

In DeepEdgeBench [5], the performance of four SoCs and a microcontroller is presented and compared in terms of inference time and energy consumption. Various models and deep learning frameworks are explored. Additionally, the energy consumption, inference time, and accuracy of the devices are measured and analyzed.

Jo et al. [6] employ various convolutional neural networks (CNNs) to analyze the performance of the Jetson Nano compared to other systems equipped with GPU acceleration, such as desktop environments and other Jetson devices.

In YOLO Benchmark [7], the operation of the You Only Look Once (YOLO) object detection model is analysed on three Single Board Computers (SBC) utilizing different accelerators: NVIDIA Jetson Nano, NVIDIA Jetson Xavier NX, and Raspberry Pi 4B (RPi) with Intel Neural Compute Stick2 (NCS2). Two different videos are evaluated in two versions of the YOLO model on the mentioned three SBCs. Additionally, the following metrics are assessed: FPS, CPU usage, memory usage, power and time.

Kang et al. [8], compare the Google Coral Dev Board and NVidia Jetson Nano devices. Additionally, they evaluate the performance of Jetson Nano in comparison with other GPU-accelerated systems. Different Convolutional Neural Network (CNN) have been used to conduct the experiment, and several metrics have been collected to facilitate the comparison.

Hui et al. [9], analyzes and compares the performance of six devices with different AI accelerators against two Convolutional Neural Networks (CNN). For this purpose, a set of metrics has been collected and utilized to determine which device is the better option for different cases in the experiment.

Díaz-de-Arcaya et al. [10], present a framework that is able to benchmark the security [11] and AI performance [12] of heterogeneous infrastructural devices in the Computing Continuum, including virtualised edge devices.

Garcia-Perez at al. [1] compare various embedded Edge devices, such as Jetson Nano, Raspberry Pi 4, and Google Coral Dev Board, against different image classification AI models. Multiple tests have been conducted on each device, involving variations in the type and size of the input images as well as AI models. Additionally, various metrics have been collected throughout the experiments.

The analysed papers present relevant insights in order to get better information about the behaviour of embedded devices, Nevertheless, as Table I highlights, all the studies identified are prepared for benchmarking ad-hoc specific scenarios with specific devices. Consequently, once the experiment has been conducted, as far as we understand, there is no possibility to execute another experiments with other features without considerable effort from developers.

As an exception, the tool developed by Diaz-de-Arcaya et al. [10] can be installed on other infrastructural devices, but the results are not portable. Furthermore, it is able to gather multiple metrics but without the possibility of selecting which ones are appropriate for the particular use case. On the other hand, the benchmarking conducted by Garcia-Perez et al. [1] features a program capable of executing AI models in a generic way. For this reason, the Artifact Diversity characteristic is partially fulfilled. Indeed, such study [1] was carried out by the majority of the authors of this article which was really helpful to identify the main challenges to face when building flexible benchmarking platforms.

Therefore, we strongly believe that, since there is no benchmarking platform covering these features, the conceptualisation and implementation of such platform would significantly enhance the market entrance of companies in this area.

III. Preliminary approach

This section presents the conceptualisation of a flexible benchmarking platform aimed at making experiments for helping companies to select the best embedded devices for a specific case or the best device composition for a specific type of ecosystem. It should be capable of dealing with experiments taking devices, metrics, artifacts and data not explicitly considered from the beginning into consideration. Thus, future experiments can be accomplished instead of only executing an ad-hoc battery of tests for a given context.

TABLE I
COMPARISON OF MAIN CHARACTERISTICS BETWEEN THE ANALYZED ARTICLES.

	Device Adaptability	Custom Metrics	Artifact Diversity	Artifact Deployment	Architecture Support
DeepEdgeBench [5]	✗	✗	✗	✗	✗
Jo et al. [6]	✗	✗	✗	✗	✗
YOLO Benchmark [7]	✗	✗	✗	✗	✗
Kang et al. [8]	✗	✗	✗	✗	✗
Hui et al. [9]	✗	✗	✗	✗	✗
Diaz-de-Arcaya et al. [10]	≈	≈	✗	✗	✗
Garcia-Perez at al. [1]	✗	✗	≈	✗	✗
BenchDL	✓	✓	✓	✓	✓

✓: completely; ≈: partially; ✗: not

The following subsections establish the key features, define the architecture and specify some applicable scenarios.

A. Key Features

The proposal of this extensible benchmarking platform offers a series of significant benefits which are materialised in the following features:

- **Adaptability and flexibility:** The platform's proposal is distinguished by its adaptability and flexibility, allowing the inclusion of devices, metrics, and diverse types of artifacts, like data transformations or models, not initially considered. Going beyond ad-hoc tests, the platform facilitates the execution of future experiments, providing greater efficiency by adapting to evolving needs and enabling the creation of specific evaluations. This capability is crucial for addressing diverse scenarios without having to foresee them, as well as to stay abreast of the continuous technological innovations.
- **Device recommendation and Analysis report:** One of the primary goals is to identify the recommend device or the best combination of devices for a specific ecosystem. This empowers users with valuable information for decision-making. The platform generates a detailed recommendation report based on criteria defined and metrics obtained during the evaluation, enhancing the user's understanding of the evaluation outcomes.
- **Efficiency, precision and rapid prototyping:** The automatic deployment of the battery test to execute, driven by a test selection component supported by an AI optimization algorithm, significantly improves the efficiency and precision of the selection process, streamlining evaluations, in addition to accelerating the benchmarking duties.
- **Inclusion of a public testing artifacts catalog:** Expanding options by incorporating various types of tests that cover a wide range of scenarios. The openness to community collaboration, facilitated by an Application Programming Interface (API), encourages contributions from third parties.

B. Architecture

The main components of the architecture are defined in this subsection. Fig. 1 illustrates the inputs and outputs, as well as the platform components. On the one hand, the input of the platform is a description of all the aspects to consider when conducting experiments. For this purpose, BenchMarkDL will be conceptualised as an extensible description language to model such aspects. In turn, a graphical editor will be implemented to quickly create BenchMarkDL-compliant documents in an usable fashion. On the other hand, the output will be a report with the device recommendation where the criteria set and metrics calculated are detailed.

Firstly, the benchmarking extensible platform parses the BenchMarkDL document using the **BenchMarkDL engine**. Then, the **Battery test intelligent selection** executes an AI optimization algorithm to select the tests that better match the input provided. The **Testing artifacts public catalogue** will be a compendium of diverse tests from where extract them and covering the widest possible scenarios, like different AI algorithms, data pipelines or resource-demanding applications. It will be publicly available to allow third-party stakeholders, like the community, provide their own artifacts by using an API. In addition, the artifacts in the catalogue will be enriched with mandatory meta-information and free tags to facilitate the matching operation and boost its discover. Subsequently, **Device and Ecosystem manager** component triggers DevOps and/or MLOps processes to sequentially deploy the selected testing artifacts into the target machines. It should be highlighted that this component will also be in charge of cleaning all the elements related with each test in order to start all the tests in the same conditions. Alongside the artifact to evaluate, specific **light agents** will also be deployed into the target machines to capture the necessary metrics. Then, those agents are responsible for populating the **Results Database** with all the measures retrieved. Finally, **the Device Recommender System** component checks the metrics obtained, the input criteria and the devices established to elaborate the output report with the device selection recommendation.

C. Applicable Domains

In the literature, numerous articles present embedded edge computing case studies applicable to multiple sectors. For example, Douch et al. [13] assert that the boost of embedded devices could improve the efficiency of sectors such as VEC (Vehicular Edge Computing), where edge computing technology enhances vehicles to be smarter and safer. Additionally, the field of OEC (Orbital or Satellite Edge Computing) is

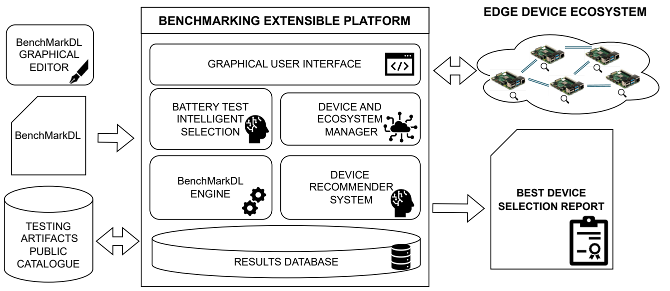

Fig. 1. BECHDL - Flexible benchmarking platform architecture overview diagram.

highlighted, aimed at equipping satellites with computing power. The concept of 'UAV-EC' (Unmanned Aerial Vehicle Edge Computing) is also addressed, involving the grouping of unmanned aerial vehicles (UAVs) to cover computing needs with low-latency connections. Furthermore, Robotics Edge Computing is mentioned, which entails the fusion of robotic resources to improve production processes and human interactions.

Additionally, Gkonis et al. [14] provide a detailed insight into how edge devices are utilized in sectors such as agriculture, energy, and maritime applications. It explains how these devices enhance efficiency and functionality in these specific areas. Hamdan et al. [15] explore how edge devices contribute to urban intelligence and improve the delivery of healthcare services. It emphasizes the relevance and impact on the creation of smarter urban environments and advanced healthcare. Finally, Cao et al. [16] offer a comprehensive overview of use cases in predictive maintenance and security monitoring using edge devices. It delves into the significance of these devices in preventing failures in industrial equipment and enhancing surveillance and security in various contexts.

IV. Conclusions and Future Work

The main objective of this paper is to conceptualize a modular and extensible reference benchmarking platform to help enterprises select a specific embedded device for a given requirement. In addition, it must be able to deal with not previously defined scenarios, contrary to ad-hoc solutions. This objective arises after analyzing the state of the art, revealing the need to have a benchmarking platform that is adaptable and flexible to different environments, distinguishing our approach from others. Furthermore, Section III presents the overall conceptualisation of this approach where, initially, the main benefits are described, then a detailed architecture and component description are proposed and, finally, several domains that can benefit from embedded edge computing are exposed to higlight the importance of building the proposed solution.

As a summary, it can be concluded that the utilisation of embedded computing can provide significant benefits to a vast number of diverse sectors. However, there is a gap to provide a benchmarking platform to help companies in their device selection as justified by the literature review and, as a consequence, to accelerate their entrance into the market.

As future work, it is expected to expand the functionalities on the proposed benchmarking platform. To do this, the description language of BenchmarkDL and the components described above will be detailed and implemented alongside a wide battery of tests to better characterise the target scenario and architectue. Moreover, AI methds will be studied to be incorporated in the core architecture. For instance, clustering techniques to better classify the devices or the tasks to accomplish.

Acknowledgment

We greatly appreciate that this work was partially financed by the European Commission through their HORIZON-CL4-2022-DATA-01 program (OASEES project, ref. 101092702).

References

[1] A. Garcia-Perez, R. Miñón, A. I. Torre-Bastida, and E. Zulueta-Guerrero, "Analysing edge computing devices for the deployment of embedded ai," *Sensors*, vol. 23, no. 23, p. 9495, 2023.
[2] Z. Zhang and J. Li, "A review of artificial intelligence in embedded systems," *Micromachines*, vol. 14, no. 5, p. 897, 2023.
[3] N. Gabdullin and A. Raskovalov, "Google coral-based edge computing person reidentification using human parsing combined with analytical method," *Internet of Things*, vol. 22, p. 100701, 2023.
[4] T. Qiu, J. Chi, X. Zhou, Z. Ning, M. Atiquzzaman, and D. O. Wu, "Edge computing in industrial internet of things: Architecture, advances and challenges," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 4, pp. 2462–2488, 2020.
[5] S. P. Baller, A. Jindal, M. Chadha, and M. Gerndt, "Deepedgebench: Benchmarking deep neural networks on edge devices," in *2021 IEEE International Conference on Cloud Engineering (IC2E)*. IEEE, 2021, pp. 20–30.
[6] J. Jo, S. Jeong, and P. Kang, "Benchmarking gpu-accelerated edge devices," in *2020 IEEE international conference on big data and smart computing (BigComp)*. IEEE, 2020, pp. 117–120.
[7] H. Feng, G. Mu, S. Zhong, P. Zhang, and T. Yuan, "Benchmark analysis of yolo performance on edge intelligence devices," *Cryptography*, vol. 6, no. 2, p. 16, 2022.
[8] P. Kang and J. Jo, "Benchmarking modern edge devices for ai applications," *IEICE TRANSACTIONS on Information and Systems*, vol. 104, no. 3, pp. 394–403, 2021.
[9] Y. Hui, J. Lien, and X. Lu, "Early experience in benchmarking edge ai processors with object detection workloads," in *International Symposium on Benchmarking, Measuring and Optimization*. Springer, 2019, pp. 32–48.
[10] J. Díaz-de Arcaya, A. I. Torre-Bastida, R. Miñón, and A. Almeida, "Orfeon: An aiops framework for the goal-driven operationalization of distributed analytical pipelines," *Future Generation Computer Systems*, vol. 140, pp. 18–35, 2023.
[11] CISOfy, "Lynis," https://cisofy.com/lynis/, 2021, last accessed 13 February 2024.
[12] A. Ignatov, "Ai benchmark," https://pypi.org/project/ai-benchmark/, 2021, last accessed 13 February 2024.
[13] S. Douch, M. R. Abid, K. Zine-Dine, D. Bouzidi, and D. Benhaddou, "Edge computing technology enablers: A systematic lecture study," *IEEE Access*, vol. 10, pp. 69 264–69 302, 2022.
[14] P. Gkonis, A. Giannopoulos, P. Trakadas, X. Masip-Bruin, and F. D'Andria, "A survey on iot-edge-cloud continuum systems: Status, challenges, use cases, and open issues," *Future Internet*, vol. 15, no. 12, p. 383, 2023.
[15] S. Hamdan, M. Ayyash, and S. Almajali, "Edge-computing architectures for internet of things applications: A survey," *Sensors*, vol. 20, no. 22, p. 6441, 2020.
[16] K. Cao, Y. Liu, G. Meng, and Q. Sun, "An overview on edge computing research," *IEEE access*, vol. 8, pp. 85 714–85 728, 2020.