Research Article

Aarne Klemetti* and Erkki Räsänen

Foundations and case studies on the scalable intelligence in AIoT domains

https://doi.org/10.1515/eng-2022-0381
received May 10, 2021; accepted October 23, 2022

Abstract: The Internet-of-things (IoT) concept is based on networked, mobile, and sensor-equipped microelectronic devices. They are capable of reacting to their environment by collecting and processing data, computing, and communicating with other IoT devices and the cloud. The deployment of artificial intelligence (AI) to IoT, referred to as artificial intelligence of things (AIoT), enables intelligent behavior for the whole cyber-physical system whether it is designed for human co-operation, completely autonomous operations, or something in between. The IoT devices, including smart phones and wearables, can be applied in a plethora of applications ranging from building automation and industrial systems to self-driving vehicles and health services. The distributed and growing usage of the connected devices deliver the users more responsive and intelligent support for decision-making in a given environment. The foundation of AI is based on data fed to algorithms for machine learning (ML). They require a lot of processing power due to the amount of data and recursive/concurrent nature of calculation. Until recently, this has been accomplished mainly in the cloud environment, where the raw data is uploaded into. This exposes all the data, even private and sensitive data, to the transmission phase and processing system. In conjunction with IoT, there is a possibility to perform ML closer to the origin of data concerning local intelligence. It means that only the results of local or edge ML are transmitted to cloud for more general aggregation of AI. Local systems do not need to send the raw data anymore, which helps on prevailing the privacy and security of the data. This type of ML is referred to as federated/collaborative learning. This study focuses on finding the existing and/or recommended solutions for up-to-date AI close to the devices. First, definitions of devices are reviewed to find out classifications of their capacity to contribute for the computation and scalability. Second, other computing and serving options between devices and the cloud are studied. Those are referred to as Fog/Edge services, and they are more stationary than the IoT devices. Third, the facts learned are being applied in two use cases to support the discussion and applicability of AIoT in practice. The main conclusion is that currently there are no single solutions – neither hardware nor software – for solving all the identified requirements were found. Instead, there are multiple options from mutually connected devices via middle-layer support to cloud services and distributed learning, respectively.

Keywords: Internet of things, artificial intelligence of things, machine learning, federated learning, edge computing, scalability

1 Introduction

When Kevin Ashton used the term Internet of things (IoT) for the first time in his presentation for Procter & Gamble in 1999, the whole concept was still waiting to mature: there were no commodity allround devices, hardware, and software available in those days. The key idea during the succeeding years was to enable intelligent behavior by low-end computers equipped with sensors and actuators. The concepts IoT and cyber-physical systems (CPS) were applied, often interchangeably. The importance of discussing these topics is in the ubiquity and accessibility of services independently of power supply, high bandwidth data transmission, computation, and data persistence.

The ecosystem around these concepts started to evolve in 2005, when the project Arduino was launched in Italy [1]. The power of Arduino's single board solution is in its lightweight, easily approachable architecture for sensor/actuator learning, experimenting, and production

* **Corresponding author: Aarne Klemetti,** Metropolia University of Applied Sciences, School of ICT, Karaportti 2, 02610 Espoo, Finland, e-mail: aarne.klemetti@metropolia.fi
Erkki Räsänen: Metropolia University of Applied Sciences, School of Smart and Clean Solutions, Leiritie 1, 01600 Vantaa, Finland, e-mail: erkki.rasanen@metropolia.fi

scale applicability. Arduino can be integrated into edge environments, but it is a microcontroller: software development is done on external systems instead of on the board.

Next step in the evolution took place in 2012, when a low-cost, general-purpose single-board computer – Raspberry Pi 1 – was released by Raspberry Pi Foundation [2]. It was a game changer due to the affordability of additional components and open source software. During the recent years, we have seen multiple high-performance devices entering the market based on the same concept. The idea with Raspberry Pi is that it contains both the properties for sensor/actuator operations and the whole development environment. No external computers of software are needed for proceeding from idea to applications.

Current edge compatible systems are equipped or can be extended with multicore 64-bit central processing units, graphics processing units, and even tensor processing units (TPUs) along with high-speed connectivity to large-capacity peripheral devices and networks. By exploiting these systems, it is possible to build scalable, high-availability (HA) clusters, not to mention the ability to run artificial intelligence (AI) and machine learning (ML) on an industrial scale on the edge.

The advent and current support of technologies for IoT, AI, and their applications like artificial intelligence of things (AIoT) have contributed to various systems from regular mobile devices to intelligent vehicles and other systems in the proximity of where the data is produced and collected from.

There is no doubt that cloud computing is the most efficient way to store large amounts of data and perform scalable computing efficiently. The cloud may not always be the primary option, though. This is because of the target system conditions, premises, response time requirements, and location. For example if the system is located in a place, where there is: 1) no connectivity to the Internet, or 2) too much latency or jitter over the communication link, and 3) high demand for security and privacy. Such systems are expected to work in remote areas independently, just like unmanned vehicles, which may operate even offshore, underground, or underwater environments. In these circumstances, the local scalable computing and data persistence services are needed. Such services are provided by edge computing systems.

The edge computing systems may contain several portable components and devices, including mobile phones, laptops, single-board computers, and intelligent vehicles, with a wide variety of connected sensors and actuators.

The term edge computing can also be referred to as fog computing. The terminology differs depending on the background of the speaker: tele communications, systems developers, or operators. From the viewpoint of this survey, differences in terminology are irrelevant.

The main focus of this article is on the AIoT provided by edge computing. The topic is approached with two use cases: first is to present the setting up of a test platform with remote building control and surveillance as practical applications, and the second one is an example of mining. The research problem is that when, where, how, and why to implement on-premises AI-related computing with mobile – or more specifically portable – computing equipment. Extracted from this problem area, we formulated the following research questions (RQs) to be answered:

(1) Which are the key properties and requirements to IoT edge computing? This is expected to lead to the answer to local data processing and persistence in the first place. The analysis of smart phones and detailed sensor/actuator technologies are left outside of this study.
(2) How to maintain the speed, reliability, safety, and security in AIoT computing? When dealing with the close to target system operations, it is important to acknowledge the trusted presence: data should be processed rapidly, efficiently, and reliably with minimizing possible leaks or intrusions.

The following is the structure of this article:

- After the introduction, we present the methods on how to address the RQs.
- Next, the literature review is presented.
- Then, the case studies are explored.
- After that the results are collected.
- Finally, the discussion and conclusions wrap up our achievements.

2 Methods

To understand the current status of AIoT, the respecting research activities needed to be mapped first. The target was to identify the trends and possible consensus on the classification of edge systems in respect to AI/ML.

The latest discussions and definitions of edge computing and AI were investigated by literature review. The scope of the selected materials was based on the relevancy in regard to the RQs and selected keywords of our topic.

After literature review, we describe and briefly analyze the commodity systems for providing the means for

AIoT in the edge. We will provide two use cases to illustrate the usability and feasibility of these systems in respect to the requirements and RQs.

3 Literature review

The purpose of this literature review was to identify the focus areas and topics concerning the research and development in edge technologies, AI, and optional implementations. We selected a collection of articles with different approach angles to this area of interest. We were looking for possible gaps between research and development, with the explanation of how to close them.

3.1 Perspectives on evolving technologies in the edge

The opportunities of doing AI in the edge is the target for Liu et al. [3]. They emphasize the challenges of computation and other resources on the edge. Their perspectives are to implement economical techniques for deep learning feasibility in low-capacity environments.

McMahan et al. [4] deliver an interesting viewpoint for federated learning (FL). The whole concept provides a safe way to reinforce the AI without passing data over the open Internet. Their results are a new solution, which they claim to reduce the need for transmitting the data and do ML on the edge in scale.

Computer vision is one of the key areas of AIoT. A survey arranged by Kittley-Davies et al. [5] point out that the visual feedback of the stages of pattern recognition is important, but difficult to achieve.

In their research, Lee and Nirjon are looking for a deep learning solution to edge computing [6]. The focus of this article is on dataset adaptation, and learning process by concentrating on feed-forward execution. The result is an effective adaptation process with efficiency.

Xiong and Chen point out the challenges on AIoT development [7]. Focusing on existing technologies, including 5G, their approach is presented with two use cases, which are considered common to edge computing. Those are streaming video analysis and industrial IoT (IIoT). The authors are in search of developing a cloud native edge computing system.

In their article, Lu and Zheng concentrate on the impact of 6G next-generation information systems [8]. They are looking forward to seeing the development of new intelligent solutions and use cases for secure interaction with systems. According to the authors, the 6G concept is in a major role as the enabling technology for different levels of connectivity and communication development.

The problem of complexity and management of IoT networks is addressed by Rafique et al. in their research on software defined networking (SDN) and edge computing [9]. Their concerns are in the different levels of vulnerabilities of the IoT systems, which could be managed by focusing on the SDN-Edge research and development.

In regard to autonomous systems, Jahan et al. [10] refer to both physical and virtual robots. Their target is in learning of security modeling in those environments. As a result, they emphasize the need for focusing the research more on possible threats and vulnerabilities on those application areas.

3.2 Viewpoints on applications

An interesting AIoT scale solution is addressed in an article concerning garbage classification. In their study, Song et al. use an applicable dataset [11]. The results reveal high accuracy for the new algorithm they developed.

A step toward low-end systems and AI is taken in a survey by Wang et al. [12]. Their aim is to implement deep learning with microcontroller units (MCUs). This kind of computation requires balancing with small memory sizes – especially considering random access memory (RAM) – and reduced computing power. With their experiments they have shown that the deep neural networks can be reduced to fit into constrained environments with acceptable performance.

A more focused approach is taken by Gao et al. [13] in a survey on the transition towards evolving intelligent robotics. That includes the next steps in collaboration between humans and robots. Also the interesting development of robot operating systems [14] is considered as a platform for more intelligent operations.

An important use case for realtime edge intelligence is unmanned aerial vehicles (UAVs). They can be extended to other areas as well, namely, underwater and terrain/subterrain operating systems. Xue et al. [15] emphasize the safety and reliability of positioning. Their solution is to use imagery collected from different

sources including the UAV's own camera. With the application of deep learning, they can detect spoofing.

Human–machine interaction (HMI) is revisited by Dong et al. [16]. They emphasize the energy supply in their research as well as the growing capacity of small-scale systems in the edge. The main conclusions they have made are the improvement of energy harvesting for increasing demand coming from more and more intricate sensors, actuators, computation, and storage solutions providing sophisticated AI for supporting decision-making.

Debauche et al. [17] introduce a new architecture for AIoT services deployment. Their approach is on application of state-of-the-art solutions: microservices, containers, AI in a customized perspective, and persistence of data with AI models. As continuation of the work by Debauche, they present also an interesting use case that is dealt with in the article about AIoT and real-time poultry monitoring. The research is carried out by Debauche et al. [18]. Their solution collects data every minute by a sensor network. The data is then applied to monitoring and prediction of conditions of poultry. An interesting finding is the way they implement a specific AI algorithm – gated recurrent unit – for environmental analysis.

Autonomous vehicles, and more specifically maritime applications, are under investigation by Chan [19]. They focus on background subtraction algorithms to provide good practices for this kind of special use cases of AIoT. As a result, they show its potential with benchmarks.

A practical approach to AIoT is taken by Zhang and Tao in their application to real-time monitoring of tunnel construction [20]. The telemetrics of operating tools in those conditions are collected and stored locally. Then the data is being applied to ML and to random forest in their case. The results show rapid responses providing real-time predictive control of construction equipment.

To the problem of environmentally sustainable systems, Yang et al. [21] propose the application of AI and more specifically reinforcement learning (RL). According to their studies, RL applied in this field of decision-making will deliver more intelligence to decision-making.

In consideration of the applicability of the IIoT Malik et al. [22] conclude in their review that applications will become ubiquitous. The foundation of their claim is in the availability enabling technologies.

Tanque [23] approach the fundamental building blocks of providing AI on IoT applications. Their research supports the usage of advanced technologies in edge-type environments to develop complete solutions from low-end data collection to AI.

4 Case studies

To concretize the AIoT and related applications, two case studies were examined from the perspective of edge computing. The first case is about the setting up of an intelligent edge computing platform following the ideas learnt from earlier studies complemented with the literature review, and experimenting with available technologies. The second case is a practical example of a production scale application of intelligent edge computing in the mining industry.

4.1 Case study 1 – Experiences on setting up a scalable AI/ML with single-board commodity development devices

The first case is about setting up an AIoT system with a target as an intelligent home environment in a rural area. The purpose of this system is to show the optional independence of the data-to-knowledge process from the cloud services. Even though the cloud is considered as an important part of these kinds of systems, the network outage should not be a show stopper. This is provided by the local operations at the edge.

Power shortage may be addressed as well by using batteries and generators. Same ideas could be applied in an industrial environment as well.

The sensors and respective data collection described here is exemplary, not meant to represent fully digitalized living conditions. It is possible that with the selected components, one might approach this environment by compiling a digital twin, with a disclaimer that not everything is being controlled, though.

The communication technology to the Internet is based on 4G with speed varying from download speed between 10 and 50 Mbps and upload speed between 1–15 Mbps. Speed depends on the load based on nearby highway, and the net activity of neighbors. In this context, the cloud-only approach is not the primary choice, given that the 4G connections can also be occasionally down. These facts motivate us to examine on-premises AIoT services, with possible cloud options, naturally.

Since this case discusses about home dependent data, it is important that following conditions are met:

- Privacy: No data nor AI models in any form may leak from the premises without consent,
- Security: If data or AI is transmitted, it should be kept in encrypted format.

- Reliability: The systems should keep data and AI persistent, even though individual components might crash or shut down.

The example system in this case is based on the following list of components (see also Figure 1 showing some components of the architecture):

- Computer development boards and other equipment:
  - 1 Raspberry Pi 4/8GB with Samsung 970 EVOPlus 500 GB SSD [24] memory device.
  - 1 Coral Google Edge TPU ML Accelerator [25] connected to the previous Raspberry.
  - 6 Raspberry Pi 4/4GB with Samsung 970 EVOPlus 500 GB SSD and 256 GB microSD memory devices: 4 are connected as a cluster; 1 as an in-house, 1 as storage building data collection stations.
  - 1 Raspberry Pi 3 with a 256 GB microSD memory card: collecting streaming video with Raspberry Pi official NoIR camera V.2 and also data from one Ruuvi multisensor [26].
  - 2 Nvidia [27] Xavier 16 GB with 256 GB microSD memory cards: one system runs a container with Timescaledb, and another container with GraphQL Application programming interface (API) between data provider, persistence, and consumers. Second Xavier 16 GB is for data science and ML development and operations.
  - Netatmo [28] weather system contains a base station, 3 additional integrated sensor stations, an anemometer, and a rain gauge.
  - 4 Arlo [29] wireless 4K cameras and 2 wireless video doorbells for surveillance of the surroundings.
  - 7 RuuviTag [26] Bluetooth Low Energy (BLE) multisensors (temperature, humidity, pressure, accelerometer, and telemetrics of the devices) measuring conditions in the refrigerator, freezer, sauna, rooms not covered with Netatmo stations, and 3 storage spaces.
  - 2 DJI Tello [30] lightweight drones with cameras and accessible APIs.
  - All devices are connected to local Intranet – wired or wireless depending on their location and equipment.
- Software tools and components
  - Operating systems for Raspberry Pis are 64-bit Raspbian versions [31]. For Nvidia devices, the operating systems are 64-bit Ubuntu versions embedded into the JetPack architecture of Nvidia [32].
  - All services are packaged into containers accessed from Docker repository, or built locally. Helm is applied for running the containers in Kubernetes [33].
  - In the cluster the containers are managed with Rancher K3S [34]. The idea here is to enable scaling out with additional different computers, but optionally to the cloud as well.

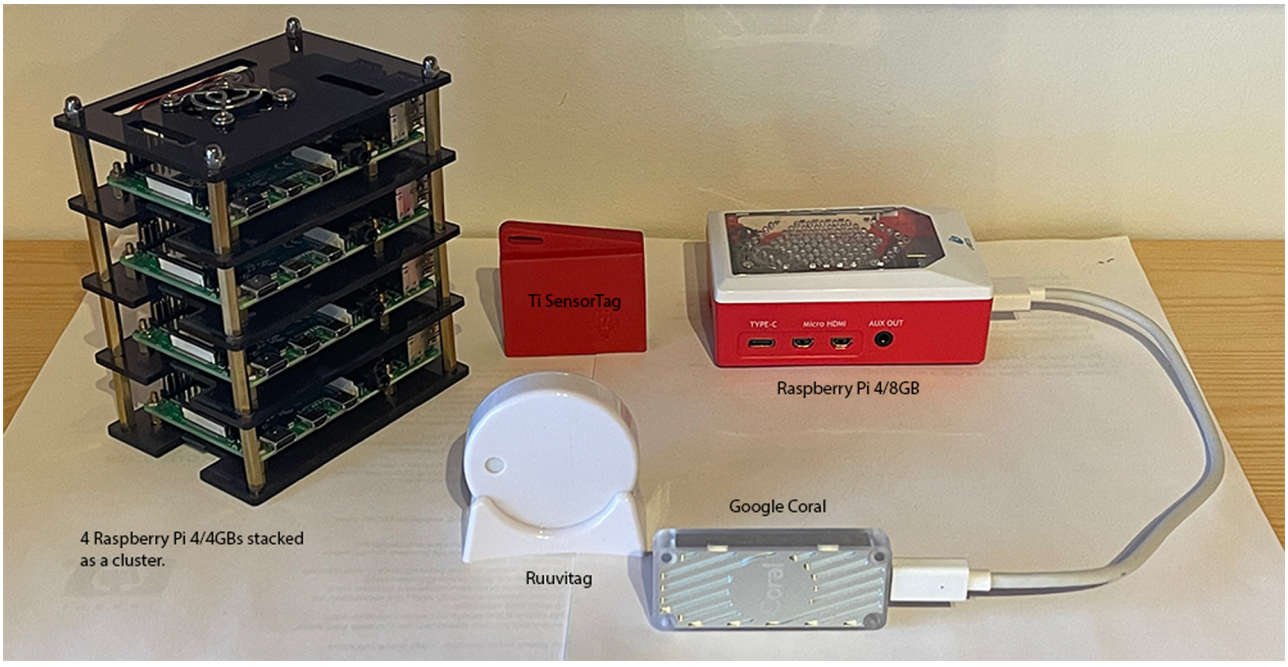

Figure 1: Example of edge system components.

- Tensorflow, Tensorflow Lite, and Tensorflow RT for deploying AI/ML [35].
- Timescaledb [36] as an industrial-scale time series database is implemented for close to real-time response times for data persistence. It is also executed in containers.
- GraphQL [37] running in its own container was selected because of its flexibility to query and mutation variations. API for dynamically changing operations with database is the target here.

Experiments carried out with aforementioned equipment:

- Computer vision for showing the possibilities of development boards in practical applications. The two camera systems have different approaches: Arlo is the manufacturer of ready-to-use entities with an option to upload videos to the cloud service. The local storage is also provided. The second system is Raspberry Pi Camera concept, where users are required to compile the devices and software by themselves. The process is presented in Figure 2.
- Local sensor data collection with AI operations as a target. The different tags containing sensors and providing appropriate APIs were put in place within the range of BLE communication with Raspberry Pis.

As a result, only demonstrations were made to learn the behavior and compatibility of software and hardware components. The different APIs were experimented, and it became clear that by splitting the activities between data collection and further processing can be done, but it needs a lot of attention and time.

It became also clear that there are APIs available for many intelligent home appliances. The camera systems from Arlo and the DJI drones provide their own connections with the options to communicate with their sensors and actuators. To set up an industrial-scale surveillance system with this equipment would require a lot more reliability and functionality of the systems.

The setup of the experimental system described earlier required many iterations. The data collection and storage processes are quite straightforward *per se*: sensors connected to nearby Raspberry Pis communicating with a relational database over a standard API technology with the support of basic operating system timing functionality.

On the other hand, a lot of the work was invested in detecting compatible software components and to make them work in practice.

Especially challenging is to make cloud-proven systems, like high availability Kubernetes to operate smoothly on the edge. Same applies to AI-related tasks, because of the novelty of all systems involved.

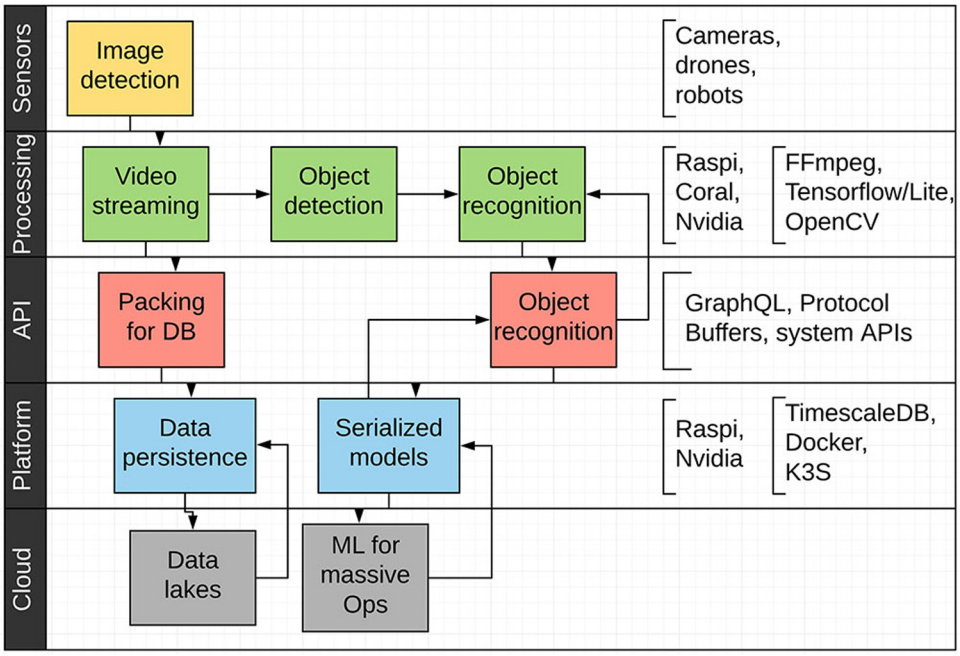

Figure 2: Image detection and respective ML process with participating devices and software tools.

4.2 Case study 2 – Water quality measurement station at a mining site

Water quality monitoring at mining sites is an excellent example of industrial IoT (IIoT) application. Water quality monitoring is required to prevent environmental emissions. IIoT technology provides a feasible way to obtain data from a sparsely distributed measurement station network.

Mining industry consumes lots of water in the refining of ore. Used process water is stored into large tailings ponds. Water is purified and then recycled back into the process (95%) and outside the mining site as effluent (5%). The quality of water is monitored, because the performance of the process depends largely on the cleanness of the recycled water, and emissions of harmful substances to nature with effluent must be prevented.

Water quality is monitored continuously at monitoring stations, which are located by the tailings ponds. Monitored parameters include electrical conductivity of water (salt concentrations), turbidity (suspended solids), pH, and flow of water. The data from the measurements are usually transferred via wireless IoT systems.

Sometimes the mining sites are located in remote areas where the availability of cellular networks and electricity is limited. These limitations set the basic requirements for the IoT edge system (i.e., the monitoring system). They usually are powered with solar cells or/and wind turbines. A radio network, e.g., LoRa, is used for local data transfer. The system operates in harsh environmental conditions – in arctic areas, the temperatures in winter are often below −35 °C.

A measurement station is shown in Figure 3. The system consists of a monitoring well module, measurement sensors, and a LoRa transmitter. The sensors are hard-wired to the transmitter via CAN bus. The system is powered with a 100 W solar panel and a wind turbine (not shown in Figure 3). The transmitter is located in an insulated cabinet. The cabinet is heated with a 50 W cabinet heater, which is just enough to keep the temperature inside above 0 °C in freezing (−35 °C) conditions.

From a measurement station, data is sent to the cloud (InFlux DB) via LoRa/Cellular router utilizing Publish-Subscribe protocols as described in Figure 4. The data can also be distributed to the production system of the

Figure 3: Water quality measurement station.

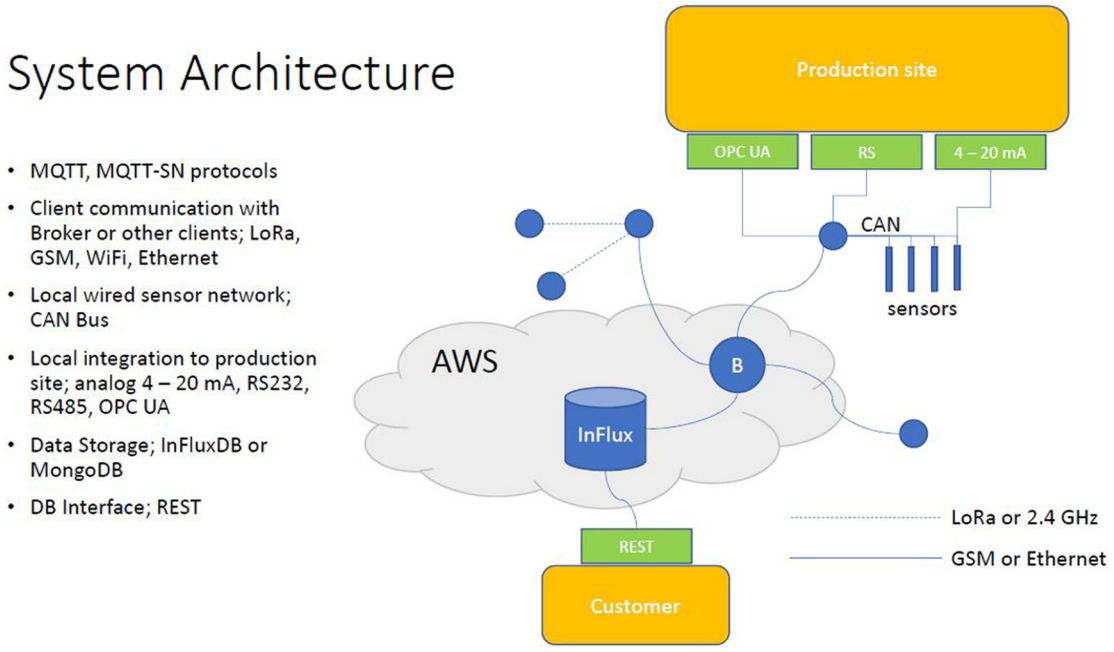

Figure 4: Water quality station's IoT framework.

mining site via the router, and the customer (mining company) has access to data stored in the database.

The quality of the data is quite critical, and the sensors are prone to so-called fouling. Foul consists of salts and dirt deposited on the surfaces of the sensors, and it interferes with the measurements. Sensors must be cleaned frequently, and there must be an indication of how much the fouling causes interference in the measurement. An ML model is used to detect the degree of fouling. This model is based on the detection of sensor signal variance and cross-correlation of all sensors with each other.

The ML model runs partially in the cloud and in the sensors as an embedded ML application. The embedded part of the ML model detects the signal variance index, while the cross-correlation of all signals is run in the cloud. Also the signal variance part could be run in the cloud, but running the model in the sensor instead of the cloud provided us a proof-of-concept of low-level embedded edge intelligence.

Referring to our RQs given earlier in the text, we may ask how the choice between cloud and edge computing should be made.

Part of the ML model requires information from all sensors, and there can be tens of monitoring stations at a mining site. Cross-correlations of all sensors against each other indicate which sensors begin to foul and should be maintained. The sensors have enough processing power to run this kind of model, but it does not make sense to send all the data to the sensors. Thus, computing the model in the cloud is justified. Alternatively, there could be an edge computer dedicated to this purpose.

On the other hand, the signal variance model does not need data from other sensors. Such a model also benefits from higher time resolution than the cross-correlation model. Thus, when running this part of the model in the sensors, we have the benefits of higher accuracy, because a high-time resolution model can be run in the processor of the sensor while keeping the data transmission rate low. This, in turn, saves the energy needed for the data transmission. Consideration of energy may sound trivial, but in arctic conditions, this matter is actually quite critical.

In this case, we have relied on both cloud and edge computing. The case is quite simple, but the benefits were well proven. With the ML model, the operators at the mining site were able to detect the sensors and stations that were in the need of maintenance. Because the access to the stations is difficult and distances long, a considerable amount of work was saved and the reliability of the water quality measurements was increased significantly.

4.3 Estimates of unit prices

The following price ranges deliver the scale of costs, since there is variability on both availability of components

and their market prices. Given the rough price ranges on the fall 2022, the estimates are following:

- Raspberry Pi: Prices vary from low end model 2 W Zero starting at 30€ to version 4/8 GB at 150€.
- Nvidia: From Jetson Nano 2GB at 50€ to Jetson AGX Orin at 2,000€,
- SSD drives: 512 GB at 70€.
- Google Coral TPU accelerator: At 145€.
- Netatmo: Full Weather Station at 380€.
- Arlo: 4 devices at 4K Ultra cameras and the station 950€.
- RuuviTag: 40€/tag.
- DJI Tello: 100€.
- The Lora-IoT base stations: at 15€/customized circuit board.
- Weatherproof cabinets and other materials at 1,000€.
- Sensors: 1,500€.

5 Results

The amount of information available on AIoT shows that the topic is interesting from the viewpoint of scientific research. The selections we made on literature review support our RQs. In our view, a more indepth survey on scientific papers should be conducted separately. This should be carried out from the viewpoints of benchmarking, testing, authorization, trust, and security of AIoT-related concepts and applicable practices in general.

The different blog articles and tutorials that came up in our searches can be used as a basis for compiling functional systems. The code snippets cannot be just copy-pasted as is, but with careful consideration of software development practices. Interestingly, there were several leads to scientific research in those articles as well. Also the endeavors to distributed computing and data persistence were prevalent in several writings.

In reflection to the RQ1, about the key properties and requirements to IoT edge computing, we can state that intelligence on the edge in the form of AIoT is possible to accomplish with generally available commodity devices. It is possible to scale in and out the systems, but one has to have a policy to do so. The two use cases show that requested systems can be accomplished independently in practically any location. The option to compile a local mini cloud system gives freedom to mission critical real-time applications to be executed on demand.

Considering the RQ2, on how to maintain the speed, reliability, safety, and security in AIoT computing the use case 1 relies on containerized HA where processing and persistence can be isolated. Use case 2 addresses the same question in practice by operating closely with the cloud and extended on-premise environment.

The implementation and delivery of the systems in two use cases requires experience on installations and configurations. Especially the setup of containers and their management required several iterations.

Important notion on the software implementation of the Raspberry Pis and Nvidia Jetson devices is that the operating system with required applications can be relatively easily changed. This is due to the fact that everything resides in the persistent storage – EEPROMs, microSDs, and SSDs in the applied hardware of this study – and the whole nature of the entity can be altered and adjusted with memory devices, re-settings, and rebooting.

Containerization also appeared to be a more flexible choice than bare metal configuration. It provides tools for scaling out with additional devices – even with heterogeneous architecture – but also an option to move over to the cloud.

6 Discussion

The two use cases in this study give arguments to consider edge computing as a feasible solution to operations including local data persistence and AI operations. The literature search supported this observation by showing that the edge technologies provide sufficient capacity in the means of computation, storage, and connectivity. A lot of concern was emphasized about data and intelligence vulnerabilities through the whole range from data collection, processing, selection, modeling, and intelligence.

One notable finding is that the definitions of edge computing are not explicit but elastic: the range from intelligent controller equipped sensors to 5G/6G base stations. One may approach the concept top down or bottom up and select their platform respectively. Especially the case of experimenting with the technologies reveal the possibility of difficulties in implementation of general purpose tools and frameworks: the software might not even compile on a selected architecture. This can be circumvented by scaling out or up to the next level – even to the cloud.

The RQs we posed were following with respectively found answers:

(1) Which are the key properties and requirements to IoT edge computing? Answer to this is that with

commodity development equipment, it is feasible to begin to work towards a minimum viable product. Simultaneously by prototyping, it is possible to gain understanding for industrial-scale system requirements.

(2) How to maintain the speed, reliability, safety, and security in AIoT computing? This question can be answered by experiments in a selected environment. Our cases show, that development should consider focused systems instead of general purpose systems, with scalability in mind as well.

In regard to energy demand, there are multiple enabling technologies for remote power supply: solar and wind energy harvesting with portable devices, lightweight generators, and batteries. In this sense, the first RQ – which are the key requirements to AIoT edge computing – is answered with acknowledgement and the list of devices in both use cases.

Edge computing is an important option not only for data collection and HA containerization but also for further operations: ML, AI, recurrent neural networks, and FL. Cloud computing should not be overlooked, though. Whenever powerful calculations are required along with scalable storage, the cloud is a viable solution. This was also the vision, when the concept of FL was originally conceived putting the cloud in the middle of mobile/edge systems to federate the locally produced and elaborated models, without transmitting sensitive data over the internet.

One should take into consideration that AI/ML processes may require a lot of data storage, communication bandwidth, and computing power. That may be alleviated with FL and by keeping the processing intensive operations in the cloud.

Our recommendation for next steps following this research are as follows:

(1) Set up experiments and benchmarks for:
   (a) Scaling in/out the operations individually, like on persistence and MLOps.
   (b) Performance on the FL applications between different edge setups and the cloud.
   (c) Endurance in simulated lock down and low-energy situations.
   (d) Optimization of the energy consumption in reference to the performance.

(2) Inclusion of production scale use cases:
   (a) Streaming data from sensors including cameras and other sensors.
   (b) Data and information fusion along with knowledge integration in real time.

7 Conclusion

In our RQs we were looking for the possibilities, challenges, and boundaries of edge computing. The focus was on considering the implementation of AI/ML processing as close to the data as possible. We needed to understand the state and focus areas in the scientific research by conducting a literature review on our problem area. We also learned during the course of using the available devices and systems that the Internet provides a wide source of valuable resources in the form of different ecosystems. We choose not to extend our literature analysis to blogs, discussion areas, frequently asked questions, and tutorials because the responsibility of proofing is left to the receiver instead of reviewers. Ecosystems generally keep things on track, though.

Our two RQs on the requirements were answered, but they need to be elaborated in every new implementation. We recommend also setting up customized environments for testing the systems. AI is a wide discipline with multiple possibilities. For edge purposes, the techniques like FL is a good starting point especially given the security and trust requirements.

It is important to understand that the technologies are evolving rapidly. The consequence is that the devices, hardware, and software are not upgraded and developed synchronously. With that in mind, the working systems tend to be version dependent until they are fully matured to production-scale operations. There is still a lot of work to be done before the AIoT is business as usual. One should not wait, though, because there are so many disciplines involved and steep learning curves around.

Acknowledgments: This paper and the research behind of it could not have been done without the reflection, support and contents of following projects, their experts, our colleagues, and participating organizations:

Our results have benefitted the following projects and will provide continuity to next phases of research and development where edge computing and AI/ML contribute impact. This support is gratefully acknowledged:

– Digi-Flash project [38] and its successor Big-Flash project [39], which are carried out under the lead of project manager Antti Liljaniemi from Metropolia UAS, and contributed by the city of Vantaa, European Regional Development Fund (ERDF) and numerous innovative participating companies.
– Hippa – Wellbeing and better service housing through digitalization [40] and its successor Hippa Remote services for product developers to promote the housing of the elderly [41] funded by 6Aika, which is ERDF-based

strategy for sustainable urban development, and Helsinki-Uusimaa Regional Council. Both projects are managed by Dr. Toini Harra from Metropolia UAS.
– EIT Raw Materials, SERENE Project, and Business Finland Project "Water Quality Monitoring System for Large Scale Monitoring Applications" [42].

Author contributions: Aarne Klemetti: Conceptualization, Methodology, Visualization, Writing - original draft, Writing – review & editing. Erkki Räsänen: Methodology, Writing – original draft, Visualization.

Conflict of interest: The authors state no conflict of interest.

Data availability statement: All the data is already in the article.

References

[1] Arduino. Wikipedia. Accessed Oct 10, 2022. [Online]. https://en.wikipedia.org/wiki/Arduino.
[2] Raspberry. Wikipedia. Accessed Oct 10, 2022. [Online]. https://en.wikipedia.org/wiki/Raspberry_Pi.
[3] Liu D, Kong H, Luo X, Liu W, Subramaniam R. Bringing AI To edge: From deep learning's perspective. Neurocomputing. 2022;485:297–320. https://www.sciencedirect.com/science/article/pii/ S0925231221016428.
[4] McMahan HB, Moore E, Ramage D, y Arcas BA. Federated learning of deep networks using model averaging. CoRR. 2016;abs/1602.05629: http://arxiv.org/abs/1602.05629.
[5] Kittley-Davies J, Alqaraawi A, Yang R, Costanza E, Rogers A, Stein S. Evaluating the effect of feedback from different computer vision processing stages: a comparative lab study. In: Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems. CHI '19. New York, NY, USA: Association for Computing Machinery; 2019. p. 1–12. doi: 10.1145/3290605.3300273.
[6] Lee S, Nirjon S. Learning in the wild: when, how, and what to learn for on-device dataset adaptation. In: Proceedings of the 2nd International Workshop on Challenges in Artificial Intelligence and Machine Learning for Internet of Things. AIChallengeIoT '20. New York, NY, USA: Association for Computing Machinery; 2020. p. 34–40. doi: 10.1145/3417313.3429382.
[7] Xiong J, Chen H. Challenges for building a cloud native scalable and trustable multi-tenant AIoT platform. In: Proceedings of the 39th International Conference on Computer-Aided Design. ICCAD '20. New York, NY, USA: Association for Computing Machinery; 2020. doi: 10.1145/3400302.3415756.
[8] Lu Y, Zheng X. 6G: A survey on technologies, scenarios, challenges, and the related issues. J Ind Inf Integr. 2020;19:100158. https://www.sciencedirect.com/science/article/pii/S2452414X20300339.
[9] Rafique W, Qi L, Yaqoob I, Imran M, Rasool RU, Dou W. Complementing IoT services through software defined networking and edge computing: a comprehensive survey. IEEE Commun Surv Tutor. 2020 thirdquarter;22(3):1761–804.
[10] Jahan F, Sun W, Niyaz Q, Alam M. Security modeling of autonomous systems: a survey. ACM Comput Surv. 2019 Sep;52(5):1–34. doi: 10.1145/3337791.
[11] Song F, Zhang Y, Zhang J. Optimization of CNN-based garbage classification model. In: Proceedings of the 4th International Conference on Computer Science and Application Engineering. CSAE 2020. New York, NY, USA: Association for Computing Machinery; 2020. doi: 10.1145/3424978.3425089.
[12] Wang Z, Wu Y, Jia Z, Shi Y, Hu J. Lightweight run-time working memory compression for deployment of deep neural networks on resource-constrained MCUs. In: Proceedings of the 26th Asia and South Pacific Design Automation Conference. ASPDAC '21. New York, NY, USA: Association for Computing Machinery; 2021. p. 607–14. doi: 10.1145/3394885.3439194.
[13] Gao Z, Wanyama T, Singh I, Gadhrri A, Schmidt R. From industry 4.0 to robotics 4.0 – a conceptual framework for collaborative and intelligent robotic systems. Proc Manufact. 2020;46:591–9, 13th International Conference Interdisciplinarity in Engineering, INTER-ENG 2019, 3-4 October 2019, Targu Mures, Romania. https://www.sciencedirect.com/science/article/pii/S235197892030963X.
[14] Robot Operating System. Wikipedia. Accessed Oct 10, 2022. [Online]. https://en.wikipedia.org/wiki/Robot_Operating_System.
[15] Xue N, Niu L, Hong X, Li Z, Hoffaeller L, Pöpper C. DeepSIM: GPS spoofing detection on UAVs using satellite imagery matching. In: Annual Computer Security Applications Conference. ACSAC '20. New York, NY, USA: Association for Computing Machinery; 2020. p. 304–19. doi: 10.1145/3427228.3427254.
[16] Dong B, Shi Q, Yang Y, Wen F, Zhang Z, Lee C. Technology evolution from self-powered sensors to AIoT enabled smart homes. Nano Energy. 2021;79:105414. https://www.sciencedirect.com/science/article/pii/S2211285520309915.
[17] Debauche O, Mahmoudi S, Mahmoudi SA, Manneback P, Lebeau F. A new edge architecture for AI-IoT services deployment. Procedia Comp Sci. 2020;175:10–19. The 17th International Conference on Mobile Systems and Pervasive Computing (MobiSPC), The 15th International Conference on Future Networks and Communications (FNC). The 10th International Conference on Sustainable Energy Information Technology. https://www.sciencedirect.com/science/article/pii/S1877050920316859.
[18] Debauche O, Mahmoudi S, Mahmoudi SA, Manneback P, Bindelle J, Lebeau F. Edge computing and artificial intelligence for real-time poultry monitoring. Proc Comp Sci. 2020;175:534–41. The 17th International Conference on Mobile Systems and Pervasive Computing (MobiSPC), The 15th International Conference on Future Networks and Communications (FNC),The 10th International Conference on Sustainable Energy Information Technology. https://www.sciencedirect.com/science/article/pii/S1877050920317762.
[19] Chan YT. Comprehensive comparative evaluation of background subtraction algorithms in open sea environments. Comput Vision Image Understanding. 2021;202:103101. https://www.sciencedirect.com/science/article/pii/S1077314220301284.

[20] Zhang J, Tao D. Empowering things with intelligence: a survey of the progress, challenges, and opportunities in artificial intelligence of things. IEEE Internet Things J. 2021;8(10):7789–817.
[21] Yang T, Zhao L, Li W, Zomaya AY. Reinforcement learning in sustainable energy and electric systems: a survey. Ann Rev Control. 2020;49:145–63. https://www.sciencedirect.com/science/article/pii/S1367578820300079.
[22] Malik PK, Sharma R, Singh R, Gehlot A, Satapathy SC, Alnumay WS, et al. Industrial internet of things and its applications in industry 4.0: State of the art. Comput Commun. 2021;166:125–39. https://www.sciencedirect.com/science/article/pii/S0140366420319964.
[23] Tanque M. Chapter 2 – Knowledge representation and reasoning in AI-based solutions and IoT applications. In: Kaur G, Tomar P, Tanque M, editors. Artificial intelligence to solve pervasive Internet of things issues. United Kingdom: Academic Press; 2021. p. 13–49. https://www.sciencedirect.com/science/article/pii/B9780128185766000022.
[24] 970 EVO Plus NVMeRRRRRR M.2 SSD 500GB. Accessed Oct 10, 2022. [Online]. Available from: https://www.samsung.com/us/computing/memory-storage/solid-state-drives/ssd-970-evo-plus-nvme-m-2-500gb-mz-v7s500b-am/#benefits.
[25] Build beneficial and privacy preserving AI. Accessed Oct 10, 2022. [Online]. Available from: https://coral.ai.
[26] Measure your world with Ruuvi Sensors. Accessed Oct 10, 2022. [Online]. Available from: https://ruuvi.com.
[27] Nvidia Jetson. Wikipedia. Accessed Oct 10, 2022. [Online]. Available from: https://en.wikipedia.org/wiki/Nvidia_Jetson.
[28] Measure your environment. Accessed Oct 10, 2022. [Online]. Available from: https://www.netatmo.com/en-us/weather.
[29] Coverage at every single angle, outside and indoors. Accessed Oct 10, 2022. [Online]. Available from: https://www.arlo.com/en-us/cameras.
[30] Tello specs. Accessed Oct 10, 2022. [Online]. Available from: https://www.ryzerobotics.com/tello/specs.
[31] Raspberry Pi OS. Wikipedia. Accessed Oct 10, 2022. [Online]. Available from: https://en.wikipedia.org/wiki/Raspberry_Pi_OS.
[32] Nvidia Jetpack. Wikipedia. Accessed Oct 10, 2022. [Online]. Available from: https://developer.nvidia.com/embedded/jetpack.
[33] Kubernetes. Wikipedia. Accessed Oct 10, 2022. [Online]. Available from: https://en.wikipedia.org/wiki/Kubernetes.
[34] Lightweight certified Kubernetes with Rancher. Accessed Oct 10, 2022. [Online]. Available from: https://rancher.com/products/k3s/.
[35] Tensorflow. Wikipedia. Accessed Oct 10, 2022. [Online]. Available from: https://en.wikipedia.org/wiki/TensorFlow.
[36] PostgreSQL for time-series. Accessed Oct 10, 2022. [Online]. Available from: https://www.timescale.com.
[37] GraphQL. Wikipedia. Accessed Oct 10, 2022. [Online]. Available from: https://en.wikipedia.org/wiki/GraphQL.
[38] Digi-Flash projects involve know-how, competitiveness and networks. Accessed Oct 10, 2022. [Online]. Available from: https://digisalama.metropolia.fi (Finnish).
[39] Digi-Flash projects involve know-how, competitiveness and networks. Accessed Oct 10, 2022. [Online]. Available from: https://bigflash.metropolia.fi/ (Finnish).
[40] HIPPA – Wellbeing and better service housing through digitalisation. Accessed Oct 10, 2022. [Online]. Available from: https://hippa.metropolia.fi/en/.
[41] Hippa Remote – Remote services for product developers to promote the housing of the elderly. Accessed Oct 10, 2022. [Online]. Available from: https://www.metropolia.fi/fi/tutkimus-kehitys-ja-innovaatiot/hankkeet/hippa-remote (Finnish).
[42] SERENE. Accessed Oct 10, 2022. [Online]. Available from https://eitrawmaterials.eu/project/serene/.