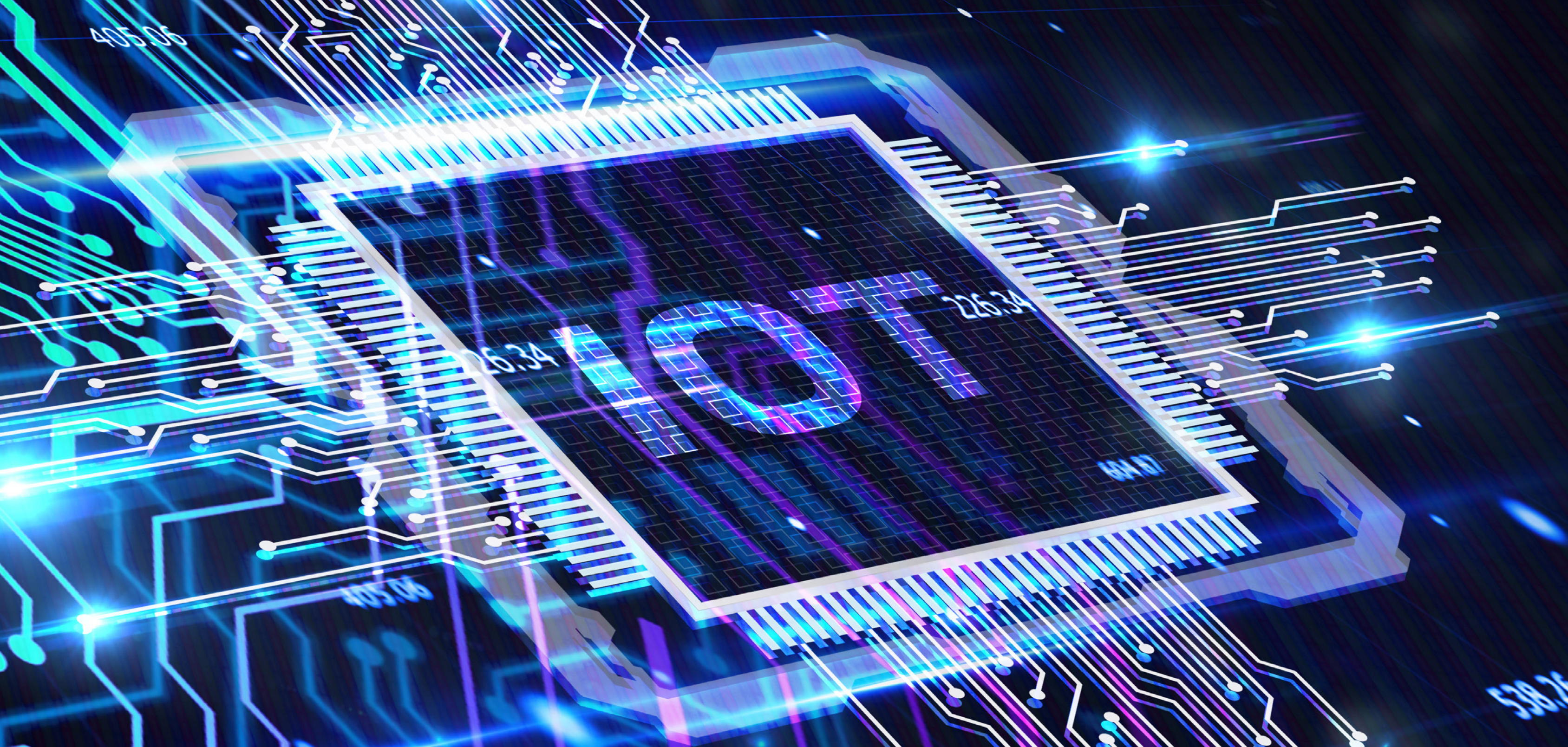
RENESAS

BRINGING INTELLIGENCE TO THE EDGE

Designing Industrial IoT Applications with TinyML

MOUSER ELECTRONICS



Foreword

MO DOGAR | VICE PRESIDENT RENESAS

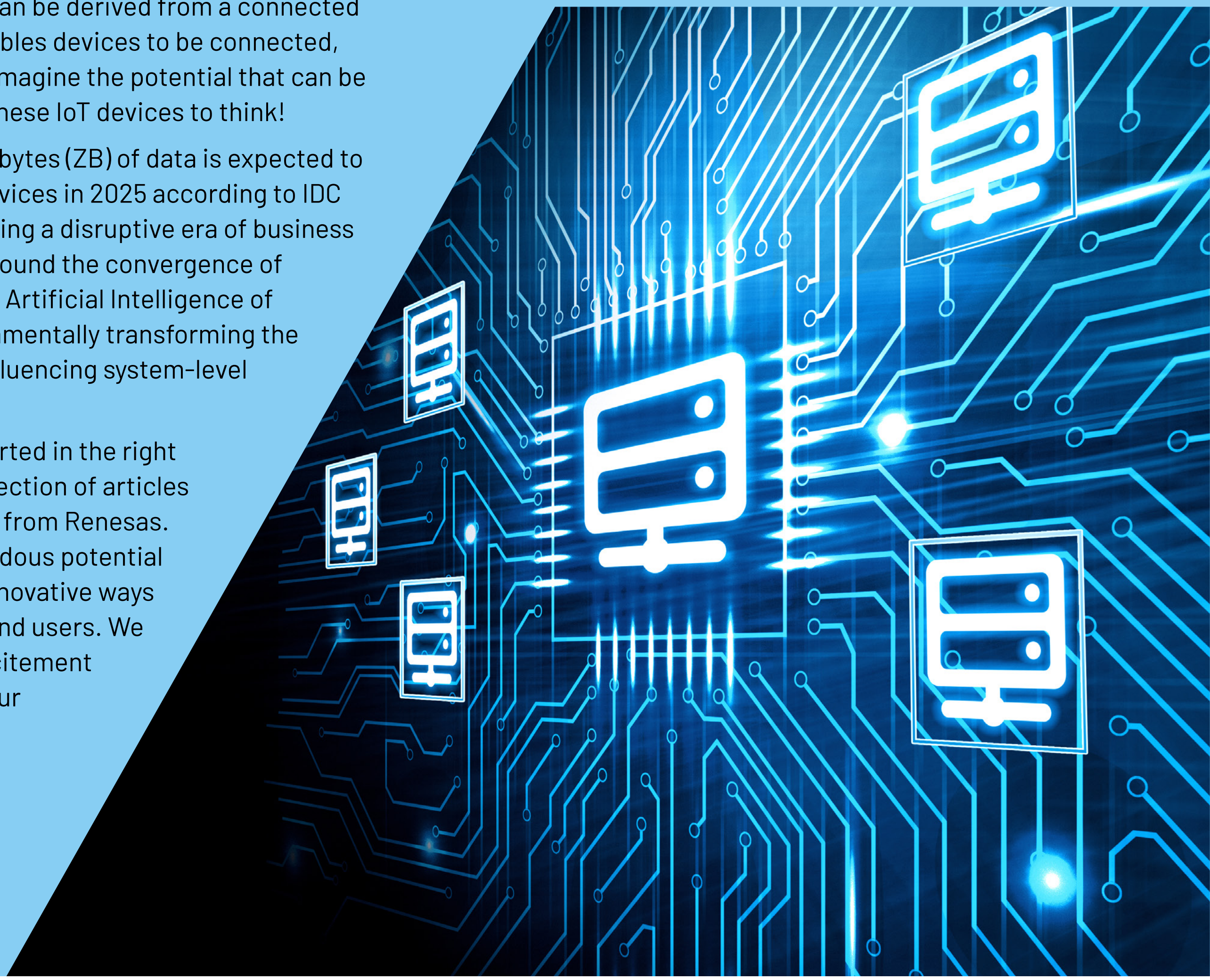
Contents

ChatGPT has shown us the tremendous potential of large language models when trained properly, and we are just getting started. What is equally fascinating is the other extreme, where artificial intelligence (AI) and machine learning (ML) are built to run within highly constrained environments, a field now known as TinyML.

We all know the potential of the Internet of Things (IoT) and the benefits that can be derived from a connected network. While IoT enables devices to be connected, AI gives them a brain. Imagine the potential that can be unlocked by enabling these IoT devices to think!

A staggering 73.1 zettabytes (ZB) of data is expected to be generated by IoT devices in 2025 according to IDC research. We are entering a disruptive era of business transformation built around the convergence of AI and IoT—or AIoT, the Artificial Intelligence of Things. AI is also fundamentally transforming the design mindset and influencing system-level approaches.

We hope to get you started in the right direction with this collection of articles and product highlights from Renesas. AI and ML have tremendous potential to solve problems in innovative ways and provide value for end users. We hope you share our excitement and make us part of your innovation journey.



AI in IoT Endpoints

With the introduction of AI, IoT devices can become more intelligent and less reliant on external systems—but not without trade-offs in performance and cost. Understanding how to make that decision is key.

BECKS SIMPSON | FOR MOUSER ELECTRONICS

Artificial intelligence (AI) is transforming the ever-expanding Internet of Things (IoT) quickly as manufacturers rush to embed the technology into their devices, from smart watches and home monitors to self-driving cars and manufacturing robots. Using AI in IoT endpoints enables powerful devices that can make decisions without the need for outside assistance, but this comes with certain trade-offs. To determine when AI should be introduced to IoT endpoints, design engineers should carefully assess the return on investment (ROI), evaluate the feasibility of the project within existing constraints, and consider data requirements and what data engineering may be needed. Embracing the right strategy when considering the costs and benefits of introducing AI to an IoT device can help to ensure the success of the project and maximize the impact of smart devices in our increasingly digitized world.

Making Smart Devices Smarter—but Not without Trade-Offs

AI is revolutionizing the way that smart devices operate, enabling them to make decisions without relying on external systems for input. This is beneficial in many ways—the data can stay on the device, which is safer and more private. The devices are also capable of making decisions based on the data without the need for immediate connectivity, which is helpful for endpoints that may have only sporadic access to a connection. However, this comes with trade-offs: AI models can be slower to run and consume more power depending on the model and size. This can be a major limiting factor for AI on edge devices because for machine learning (ML) to be effective in those applications, it must be small and fast enough to run on an edge device. Therefore, although AI-enabled devices can be more powerful, they come with trade-offs that engineers must consider.

Balancing Improved Intelligence with Impact on Performance

Design engineers should consider the ROI when deciding whether to add AI to an IoT endpoint. Before embarking on the decision to embed ML in these endpoints, the expected value should be defined and quantified as much as possible. Doing so helps not only to understand how the costs stack up compared to expected gains but also to inform the criteria for success of the project (i.e., accuracy of the model or some other metric). This will make it easier to determine if the value of the AI outweighs the impact on the device's performance.

In most cases, ML algorithms require more computing power and more energy, which can reduce the device's battery life or noticeably degrade its performance in terms of latency. If the device is already running at maximum capacity, the ML algorithms may not be able to run at a satisfactory level. Thus, design engineers should examine the use case of the device carefully to determine if AI would be beneficial and not too much of an additional strain on the device's resources. For devices that have constant access to a power source, the additional cost of computation at the source becomes more of an issue than battery life.

In addition to evaluating ROI versus device performance, design engineers should examine whether the task can be achieved with ML. Given the hype around AI, applying it to all use cases without thorough investigation is tempting.

“AI models can be slower to run and consume more power depending on the model and size.”

However, conducting research on prevailing methods in the field and applications that continue to elude satisfactory solutions from ML could reveal potential issues before starting the process of adding AI to IoT endpoints.

At this point, design engineers should also determine if the AI capabilities can be achieved without ML and if the use case is better suited for an AI-enabled device or a simpler alternative solution. Typically, the first place to start in this process is to review the literature and identify which methods have been used successfully before. Then, using that information as a guide, build a baseline proof of concept using simple, rule-based approaches or less sophisticated algorithmic implementations. This baseline can be used to assess how well a non-ML solution might work; if it does not meet success criteria for performance, then progressing to an ML-based solution may be in order.

Assessing Feasibility among All the Constraints

Assuming that the value expected by introducing ML outweighs the costs and potential performance impacts, the next step in the decision-making process is to assess whether the ML lifecycle is possible within the current technical or physical constraints of the devices. IoT endpoints that are not normally Wi-Fi enabled and that produce vast quantities of unlabeled data stored for only short periods of time at the edge will score much lower on the feasibility scale than those with cloud access, where data can be pooled and annotated more easily and where compute power is more readily available. In particular, since AI generally requires some training that involves the model having access to enough data to learn from, how this training will be done is an aspect of feasibility. For example, putting ML in an endpoint, though not impossible, is significantly less feasible if the data are distributed and cannot be pooled for training a model owing to privacy, storage, or connectivity requirements and constraints. In these cases, training could occur in a federated fashion, but the cost and complexity of doing so could outweigh the benefits.

The impact of these constraints often extends beyond the initial data annotation and model training phases of the ML lifecycle. Model performance monitoring and updates are key, requiring retraining on new data if performance has declined as well as connectivity to push new deployments to devices. Engineers should consider monitoring in terms of the model and the data: Assessing feasibility includes determining whether data drift between the training data and new points being measured can be captured. For example, monitoring is more difficult for endpoints where data are stored only for a short period of time, which may impact the final decision about whether to introduce ML to the process.

Addressing the Age-Old Question of Data

Regardless of the final application, when deciding whether to embark on an ML project, one of the most common questions is whether enough data exist. This goes hand in hand with labeling requirements—for example, whether the task requires annotated data for a supervised training approach and whether the data are currently in the required state. Within the IoT context, if the data cannot be pooled in the cloud, design engineers must determine whether enough data exist per device for the task at hand and if they need to be labeled. Data quality is essential for successful ML projects, so engineers should also evaluate the ability to manage noise and the heterogeneity of sensor data to an acceptable standard.

Data engineering depends on the device configuration and how the data are stored. To ensure the success and accuracy of an ML project, proper data engineering is essential and IoT applications are no exception. This includes understanding the data format, quality, and type, as well as how the data might be cleaned (e.g., granting access for initial data exploration and deploying processes that transform the data to be usable). Data engineering is typically performed by data engineers who require access to the data they need to work with; in cases where access is not permissible due to device location, lack of connectivity, or privacy concerns, the feasibility of project success may decrease.

Conclusion

When deciding whether to introduce AI to IoT endpoints, design engineers must take a variety of considerations into account. From assessing the expected value of the AI versus the cost and impact on device performance, to evaluating the feasibility of the ML lifecycle within the existing constraints, and finally to understanding the data requirements and what data engineering might be needed, engineers should carefully consider each element of the decision. With the right approach and research, design engineers can make an informed decision that incorporates the costs and benefits of introducing AI to an IoT device, helping to ensure the success of the project.

Revolution of Endpoint AI in Embedded Vision Applications

SUAD JUSUF | SENIOR MANAGER RENESAS ELECTRONICS

Endpoint AI is a new frontier in the space of artificial intelligence (AI) that takes the processing power of AI to the edge. It is a revolutionary way of managing information, accumulating relevant data, and making decisions locally on a device. Endpoint AI employs intelligent functionality at the edge of the network; in other words, it transforms the Internet of Things (IoT) devices that are used to compute data into smarter tools embedded with AI features. This in turn improves real-time decision-making capabilities and functionalities. The goal is to bring machine-learning-based, intelligent decision-making physically closer to the source of the data. In this context, embedded vision shifts to the endpoint. Embedded vision incorporates more than breaking down images or videos into pixels—it is the means to understand pixels, make sense of what is inside, and support making a smart decision based on specific events that transpire. There have been massive endeavors at the research and industry levels to develop and improve AI technologies and algorithms.

What Is Embedded Vision?

Embedded computer vision is a technology that imparts machines with the sense of sight, which enables them to explore the environment with the support of machine-learning and deep-learning algorithms. There are numerous applications across several industries whose functionality relies on computer vision, thus becoming an integral part of technological procedures.

In precise terms, computer vision is one of the AI fields that enables machines to extract meaningful information from digital multimedia sources to take actions or make recommendations based on the information that has been obtained. Computer vision is, to some extent, akin to the human sense of sight. However, the two differ on several grounds. Human sight has the exceptional ability to understand many and varied things from what it sees. On the other hand, computer vision recognizes only what it has been trained on and what it is designed to do exactly, and that too with an error rate. AI in embedded vision processes trains the machines to perform supposed functions with the least processing time and has an upper edge over human sight in analyzing hundreds of thousands of images in a lesser timeframe.

Embedded vision is one of the leading technologies, with embedded AI used in smart endpoint applications in a wide range of consumer and industrial applications. There are a number of value-added use cases examples, such as counting/analyzing the quality of products on a factory line, keeping a tally of people in a crowd, identifying objects, and analyzing the contents of a specific area in the environment.

While considering the processing of embedded vision applications at the endpoint, the performance of such an operation may face some challenges. The data flow from the vision-sensing device to the cloud for the purposes of analyzing and processing could be very large and may exceed the network available bandwidth. For instance, a 1920×1080px camera operating with 30 frames per second (FPS) may generate about 190MB/s of data. In addition to privacy concerns, this substantial amount of data contributes to latency during the round trip of data from the edge to the cloud, then back again to the endpoint. These limitations could negatively impact the employment of embedded vision technologies in real-time applications.

IoT security is also a concern in the adoption and growth of embedded vision applications across any segment. In general, all IoT devices must be secured. A critical issue and concern in the use of smart vision devices is the possible misuse of sensitive images and videos. Unauthorized access to smart cameras, for example, is not only a breach of privacy, but it could also pave a way for a more harmful outcome.

Vision AI at the Endpoint

- Endpoint AI can enable image processing to infer a complex insight from a huge number of captured images.
- AI uses machine-learning and deep-learning capabilities within smart imaging devices to check a huge amount of previously well-known use cases.
- For optimum performance, embedded vision requires AI algorithms to run on the endpoint devices and not transmit data to the cloud. The data here is captured by the imaging recognition device, then processed and analyzed in the same device.

In applications where power consumption at the endpoint is limited, microcontrollers (MCUs) or microprocessors (MPUs) need to be more efficient to take on the high volumes of multiply-accumulate (MAC) operations that are required for AI processing.

Deployment of AI Vision Applications

There are unlimited use cases for the deployment of AI in vision applications in the real world. Here are some of the examples where Renesas Electronics can provide comprehensive MCU- and MPU-based solutions, including all the necessary software and tools to enable quick development.

Smart Access Control

Security access control systems are becoming more valuable with the addition of voice and facial recognition features. Real-time recognition requires embedded systems with very high computational capabilities and on-chip hardware acceleration. To meet this challenge, Renesas provides a choice of MCU or MPU that offers very high computational power that also integrates many key features that are critical to high-performance facial and voice recognition systems, such as built-in H.265 hardware decoding, 2D/3D graphic acceleration, and error correction code (ECC) on internal and external memory to eliminate soft errors and allow for high-speed video processing.

Industrial Control

Embedded vision has a huge impact as it enhances many applications, including product safety, automation, and product sorting. AI techniques can perform multiple operations in the production process (such as packaging and distribution), which can ensure quality and safety during production in all stages. Safety is needed in areas such as critical infrastructure, warehouses, production plants, and buildings that require a high level of human resources.

Transportation

Computer vision presents a large scale of ways to improve transportation services. For example, in self-driving cars computer vision is used to detect and classify objects on the road. It is also used to create 3D maps and estimate movement around. By using computer vision, self-driving cars gather information from the environment using cameras and sensors, which then interpret and analyze the data to make the most suitable response by using vision techniques such as pattern recognition, feature extraction, and object tracking (**Figure 1**).

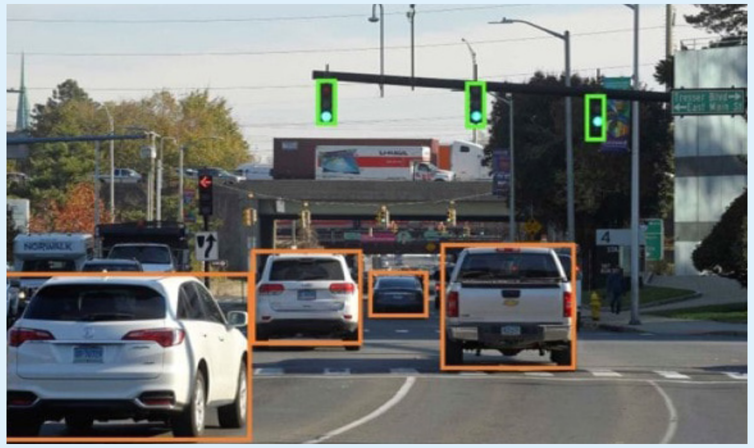


Figure 1: *Computer vision is used to detect and classify objects on the road. (Source: Renesas Electronics)*

In general, embedded vision can serve many purposes, and these functionalities can be used after customization and the needed training on different types of datasets from many areas. Functionalities include monitoring physical area, recognizing intrusion, detecting crowd density, and counting humans or objects or animals. They also include identifying people or finding cars based on license plate numbers, detecting motion, and analyzing human behavior in different cases.

> “Embedded vision is one of the leading technologies, with embedded AI used in smart endpoint applications in a wide range of consumer and industrial applications.”

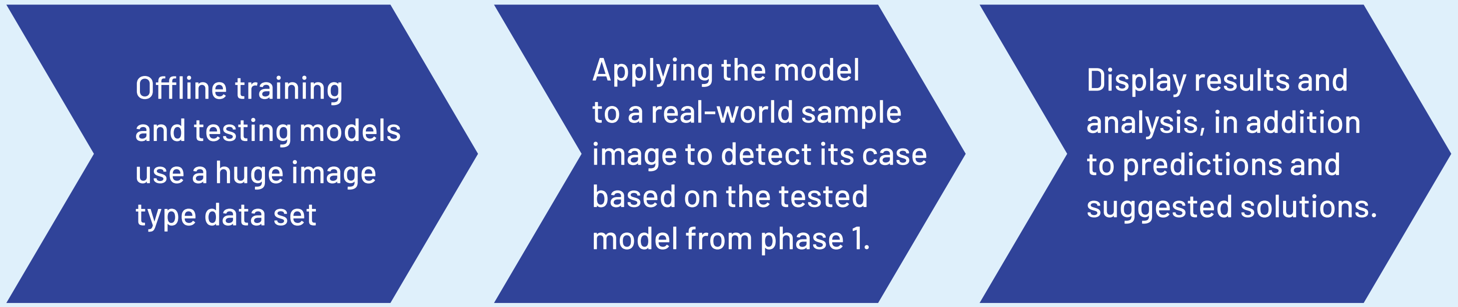


Figure 2: *Three main steps for using deep learning in computer/machine vision. (Source: Renesas Electronics)*

Case Study: Agricultural Plant Disease Detection

Vision AI and deep learning may be employed to detect various anomalies—for example, detecting plant disease. Deep learning algorithms—one of the AI techniques—are used widely for this purpose. According to research, computer vision gives better, more accurate, faster, and lower-cost results compared to the costly, slow, and labor-intensive results of previous methods. The process that is used in this case study can be applied to any other detection. There are three main steps for using deep learning in computer/machine vision (**Figure 2**).

Step one is performed on normal computers in the lab, whereas step two is deployed on a microcontroller at the endpoint, which can be on the farm. Results in step three are displayed on the screen on the user side. **Figure 3** shows the process in general.

Conclusion

We are experiencing a revolution in high-performance smart vision applications across a number of segments. The trend is well supported by the growing computational power of microcontrollers and microprocessors at the endpoints, opening up great opportunities for exciting new vision applications. Renesas Vision AI solutions can help you to enhance overall system capability by delivering embedded AI technology with intelligent data processing at the endpoint. Renesas's advanced image processing solutions at the edge are provided through a unique combination of low-power, multimodal, and multi-feature AI inference capabilities. Take the chance now and start developing your vision AI application with Renesas Electronics.

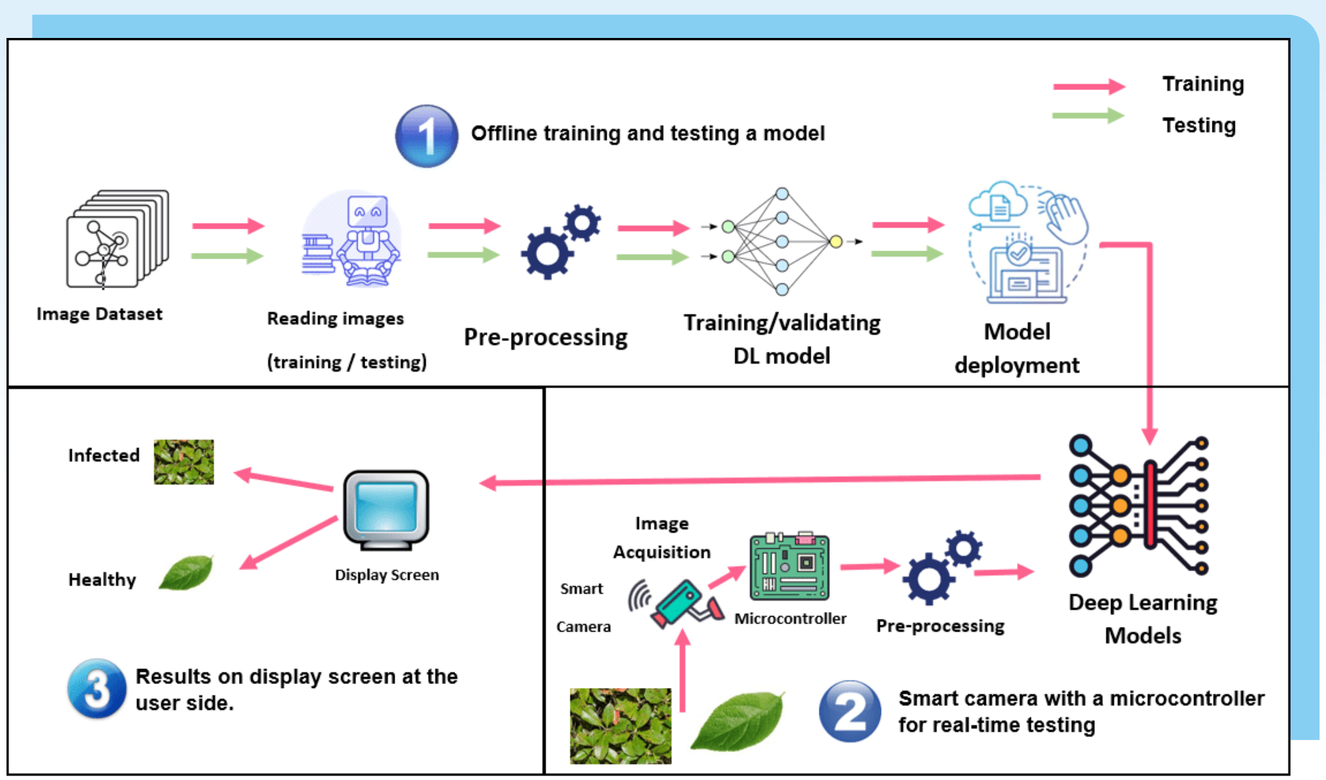


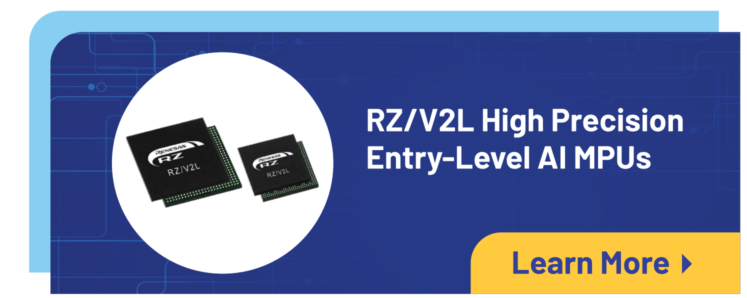
Figure 3: *General process of deep learning in computer/machine vision. (Source: Renesas Electronics)*

Vision AI Gateway Solution

Renesas Electronics' Vision AI Gateway solution efficiently handles vision information from multiple cameras and performs high-speed artificial intelligence (AI) processing. It includes a Renesas RZ/V2M AI-MPU with rich peripheral functions and an optimized power supply system. The solution realizes high performance and low power consumption at the same time. It delivers better value for gateway applications by integrating edge AI capability.

System Benefits:

- Fast vision AI processing via decoding of video streams (H.264 or H.265) and handling multiple AI inferences with minimum switching overhead
- Support for high-speed vision gateway function with Ethernet protocol: 100/1000Mbps and fast USB connection: USB 3.1 Gen 1 (5Gbps)
- Peripheral extension capabilities with PCI express 2.0 (Gen 2/2 Lane)
- Built-in power sequencing control function of the RZ/V2MA, making power supply design easier and improving reliability
- Integration of various communication devices, high-performance Wi-Fi module, compact and high-performance *Bluetooth®* Low Energy module, and LTE Cat-M1 cellular IoT module



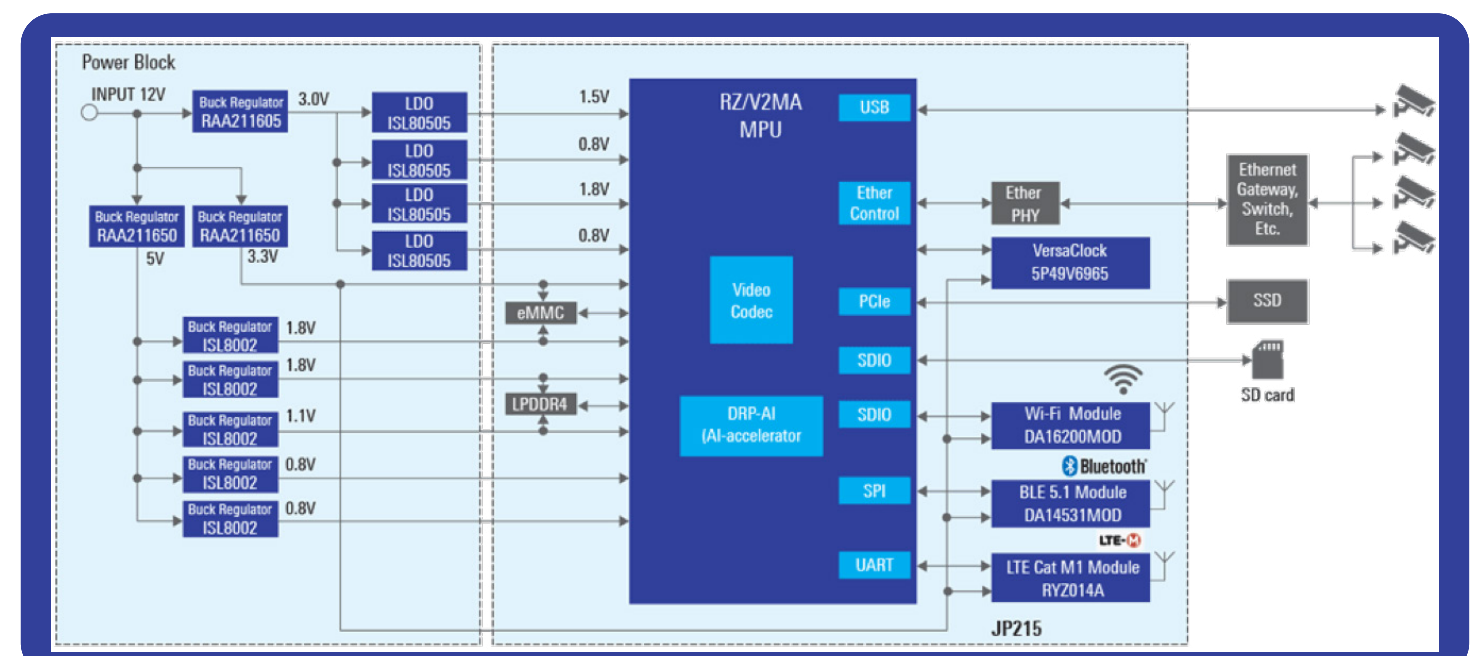


Figure 1: *Vision AI Gateway solution. (Source: Renesas Electronics)*

Implementing Machine Learning at the Edge

RENESAS
MOUSER ELECTRONICS

WHAT IS REALITY AI?

- Software tool for developing intelligent, embedded AI applications for edge devices.
- Automatically generates machine learning models, explanatory visualizations, and hardware design analytics.
- Complete frameworks for specific use cases, including hardware, firmware, software and ML reference designs.

HOW TO USE IT?

1 GATHER DATA

- Collect data from different sources.
- Upload or link source files.

2 ORGANIZE DATA

- Assign training labels.
- Segment files into consistent sample sizes for AI exploration, training, or testing.

3 EXPLORE

- Create potential feature sets and ML models for classification.
- Select the best-fit model based on accuracy and computational complexity.

4 TRAIN

- Train the new base tool against a subset of data.
- Balance is important.

5 TEST

- Test the trained tool against a subset of data.
- Validate performance using training samples and k-fold validation.

6 OPTIMIZE

- Optimize for sensitivity and tolerances.
- Retrain as needed.

7 DEPLOY YOUR MODEL

- Reality AI software comes with integration to Renesas e^2 studio.
- Compile for a wide range of Renesas processors.

WHERE TO USE IT?

- Automotive Sound Recognition (SWS)
- Industrial Anomaly Detection
- Self-diagnosing HVAC and refrigerant systems

Four Metrics You Must Consider When Developing TinyML Systems

ELDAR SIDO | PRODUCT MARKETING SPECIALIST RENESAS ELECTRONICS

Recently, with advancements in machine learning (ML) there has been a split into two scales: traditional large ML (cloud ML), with models getting larger to achieve the best performance in terms of accuracy, and the nascent field of tiny machine learning (TinyML), where models are shrunk to fit into constrained devices to perform at ultra-low power. As TinyML is a nascent field, this blog will discuss the metrics to consider when developing systems incorporating TinyML and current industry standards into benchmarking TinyML devices.

The four metrics that will be discussed are accuracy, power consumption, latency, and memory requirements. The system metric requirement will vary greatly depending on the use case being developed.

Accuracy has been used as the main metric for the performance of ML models for the last decade, with larger models tending to outperform their smaller predecessors. In TinyML systems, accuracy is also a critical metric, but a balance with the other metrics is more necessary, compared to cloud ML.

> "The system metric requirement will vary greatly depending on the use case being developed."

Power consumption is a critical consideration, as TinyML systems are expected to operate for prolonged periods on batteries (typically in the order of milliwatts). The power consumption of the TinyML model would depend on the hardware instruction sets available. For example, an Arm® Cortex®-M85 processor is significantly more energy efficient than an Arm Cortex-M7 processor, thanks to the Helium instruction set. It would also depend on the underlying software used to run the models (i.e., the inference engine); for example, using the CMSIS-NN library improves the performance drastically as compared to reference kernels.

Latency is important, as TinyML systems operate at the endpoint and do not require cloud connectivity, the inference speeds of such systems are significantly better than cloud-based systems. Furthermore, in some use cases, having ultra-high inference speed (in milliseconds) is critical to be production ready. Similar to the power consumption metric, it depends on the underlying hardware and software.

Memory is a big hurdle in TinyML, as designers squeeze down ML models to fit into size-constrained microcontrollers (less than 1MB). Thus, reducing memory requirements has been a challenge, and during model development, many techniques, such as pruning and quantization, are used. Furthermore, the underlying software plays a large role as better inference engines optimize the models more effectively (better memory management and libraries to execute layers).

As the four metrics are correlated—there tends to be an inverse correlation between accuracy and memory but a positive correlation between memory, latency, and power consumption—improving one could affect the others. So when developing a TinyML system, it is important to carefully consider this. A general rule would be to define the necessary model accuracy required as per the use case and then compare a variety of developed models against the three other metrics (**Figure 1**), given a dummy example of a variety of models that have been trained.

The marker shapes represent different model architectures with different hyperparameters, that tend to improve accuracy with an increase in architecture size at the expense of the other three metrics. Depending on the system-defined use case, a typical region of interest is shown, from that, only one model has 90% accuracy, if higher accuracy is required, the entire system should be reconsidered to accommodate the increase in the other metrics.

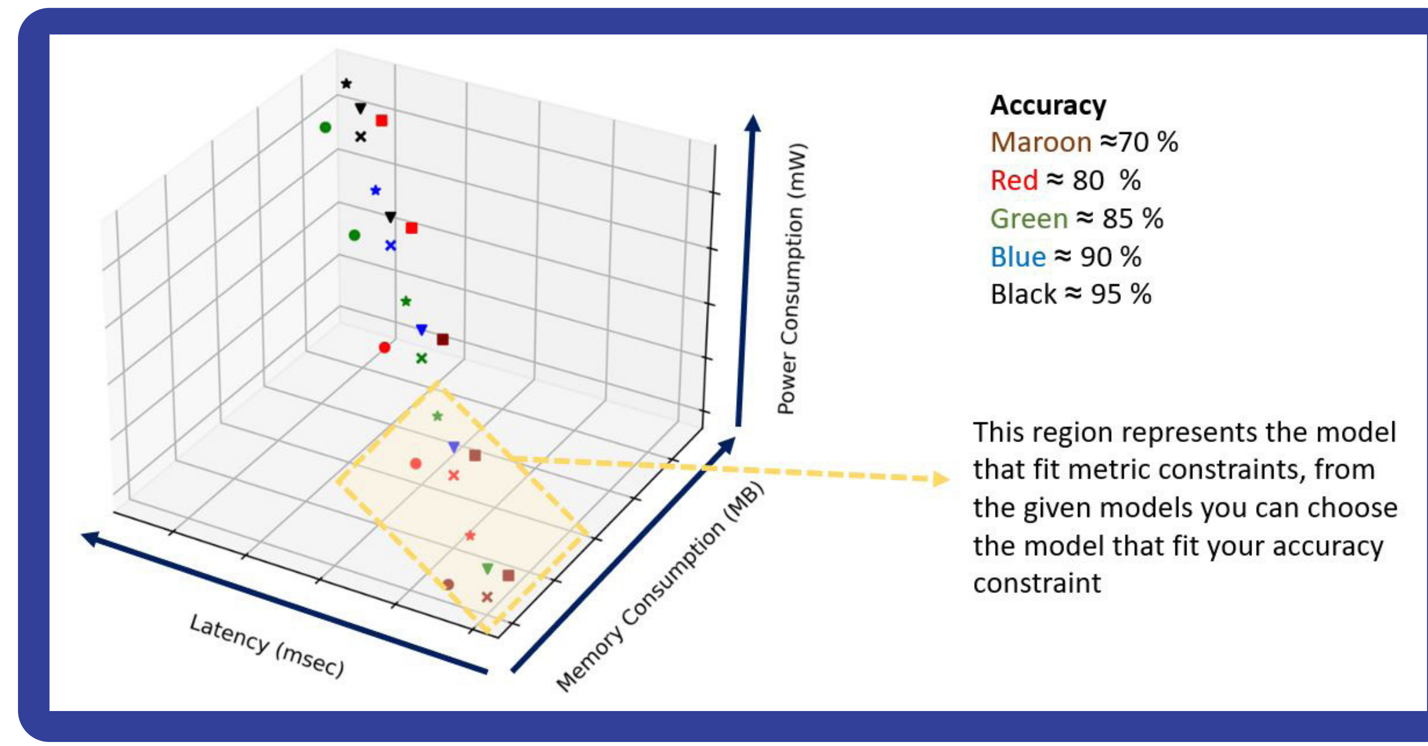


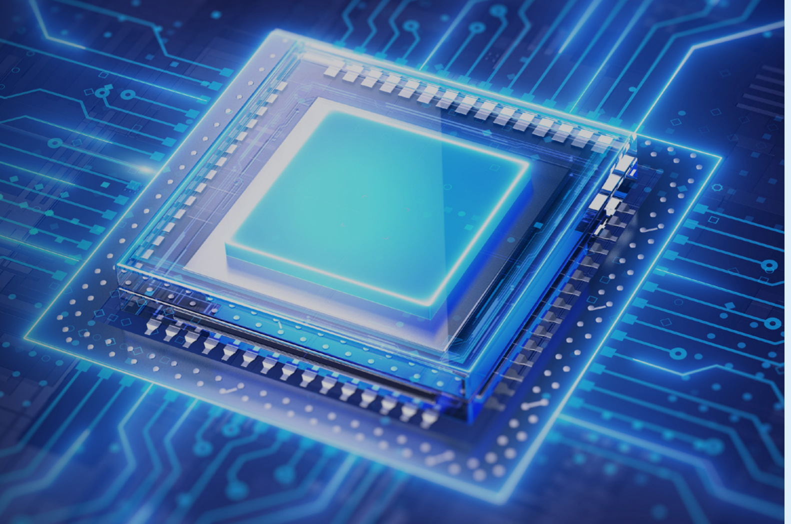
Figure 1: *Example of metrics to consider when developing systems incorporating TinyML. (Source: Renesas Electronics)*

Benchmarking TinyML Models

Benchmarks are necessary tools to set a reproducible standard to compare different technologies, architectures, software, etc. In AI/ML, accuracy is the key metric to benchmark different models. In embedded systems, common benchmarks include EEMBC's CoreMark and ULPMark, measuring performance and power consumption, respectively. In the case of TinyML, MLCommons has been gaining traction as the industry standard where the four metrics discussed previously are measured. Due to the heterogeneity of TinyML systems, to ensure fairness, four AI use cases with four different AI models are used and have to achieve a certain level of accuracy to qualify for the benchmark. Renesas benchmarked two of its microcontrollers, RA6M4 and RX65N, using TensorFlow Lite for microcontrollers as an inference engine, and the results can be viewed here.

Data Science and AI-Driven Real-Time Analytics

KAUSHAL VORA | SENIOR DIRECTOR RENESAS ELECTRONICS
SUAD JUSUF | SENIOR MANAGER RENESAS ELECTRONICS



AI is proving to be a more precise and time-efficient tool in processing the big data crunch by recognizing patterns and noticing inconsistencies in real time.

Most modern consumer-driven industries accumulate large volumes of data, but without a filtering mechanism that maps, charts, and trends data models, the raw data has little to no utility. Traditional analytical mechanisms are no longer up to the task—the sheer amount of data and the speed at which it's being accumulated has made it economically and operationally necessary to scale up analytical capacity using data science.

The Internet of Things (IoT) is constantly collecting data, and the organization, distribution, and analysis of all this data is the new frontier of artificial Intelligence (AI). While conventional data analysis methods are still being implemented, AI is proving to be a more precise and time-efficient tool in processing the big data crunch. AI can recognize patterns and notice inconsistencies in real time. AI algorithms can save significant time and energy by compiling data from multiple sources and presenting it with a uniform, consistent approach. AI and machine learning can substantively improve secure and predictive analytics operations in the data center and beyond, anticipating and even preventing potential issues before they become problems. AI can restructure data collection and analysis operations in real time and reduce unanticipated network interruptions. The marriage of AI and data science can improve data set structuring, help reach conclusions and significantly enhance IoT operations.

Data Science and Machine Learning

Data science is the integrated approach of extracting insights from growing volumes of data using various scientific methods, algorithms, and processes. Data science is driven by software that recognizes hidden patterns within unprocessed data and draws conclusions from those patterns. These valuable insights help facilitate corporate decision-making by interpreting problems analytically and generating viable solutions.

Cross-Industry Standard Process for Data Mining (CRISP-DM) is a process model that provides an overview of the data science life cycle. It's essentially a framework that assists in planning, organizing, and implementing a data science project. CRISP-DM consists of the steps in **Figure 1**:

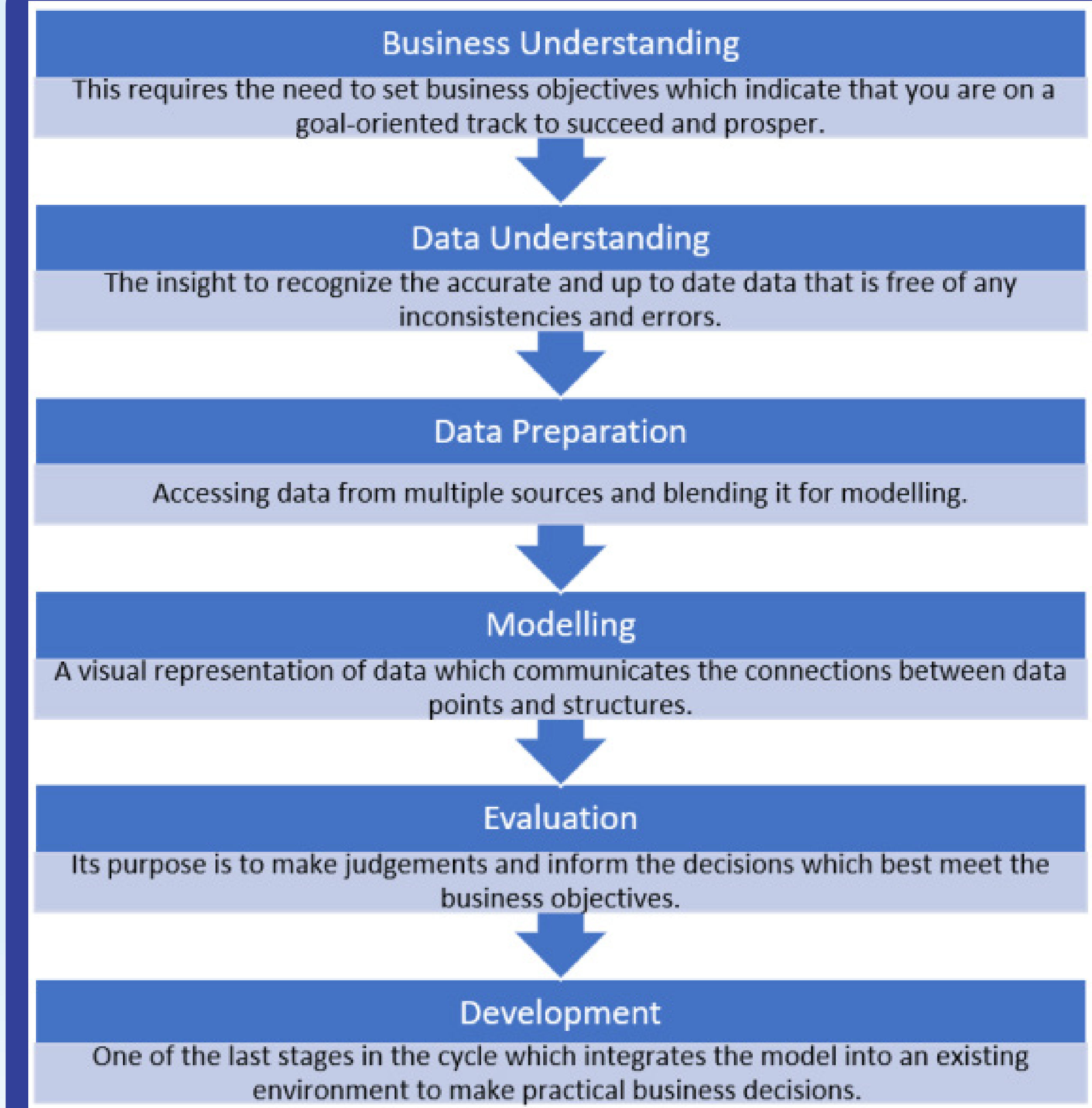


Figure 1: *CRISP-DM steps. (Source: Renesas Electronics)*

Machine learning is a branch of AI dependent on models to perform autonomous tasks without required extensive programming. It relies on statistical techniques or algorithms that allow users to make predictions and/or decisions based on historical data. Data scientists utilize technologies like machine learning and AI to examine data and make predictions about imminent developments and potential issues, significantly reducing the need for human intervention.

When critical thinking encounters machine-learning algorithms, data science can help achieve better insights, guide efficiency efforts, and inform predictions. The goal is for businesses to benefit from data science to make progressive decisions and create more innovative products and services.

What Is Data Analytics?

Data analytics is the science of scrutinizing raw or unprocessed data to derive meaning so it can be applied to decision-making. Data analytics employs several modern tools and techniques that help to structure data and make it understandable.

Several preliminary steps need to be applied in the process before data analytics can begin:

- Data needs to be collected from a variety of sources.
- Data needs to be interpreted correctly so it may be accurately organized and grouped.
- Data needs to be examined to eliminate any replications or transcription errors.
- Data may be managed by tables and/or spreadsheets.

Data analytics can be subdivided into four main categories (**Figure 2**): descriptive, diagnostic, predictive, and prescriptive.

Real-time analytics is the process of preparing, assessing, and analyzing data as it becomes available. Coupling real-time analytics with AI has provided businesses with fresh and revealing insights into the consumer experience. It has provided support personnel and IT the opportunity to act preemptively rather than react, resolving potential issues before endpoint users even realize there is an issue to report. The combination of AI and real-time analytics has completely transformed the consumer and consumer support experiences.

Endpoint Analytics of Things

According to the analytics firm International Data Corporation (IDC), 45% of data collected from IoT devices needs to be analyzed nearer to the endpoint (i.e., closer to the device itself) to optimize overall efficiency rather than transmitted to the cloud. Shifting away from the cloud has several potential advantages: Transferring data to the cloud can be costly, as it requires significant bandwidth and power to support large and potentially lengthy data transmissions. Potential latency issues are also a factor; as the time expended during transmission increases, so does the need for higher server capacities to handle all incoming data. Situations may also arise where IoT devices lack a stable internet connection, and decisions must be made at the endpoint. Solutions that can support the consumer experience using analytics at the endpoint are often required.

Descriptive Analytics	This is the process where data is employed to examine, understand, and describe an event that's already transpired.
Diagnostic Analytics	This goes deeper than descriptive analytics as it pursues to comprehend the why behind what happened.
Predictive Analytics	It is dependent on historical data, past trends, and assumptions to answer projective questions about the future
Prescriptive Analytics	This process aims to identify specific actions that an individual or organization should take to reach future targets or goals

Figure 2: *Four main categories of data analytics. (Source: Renesas Electronics)*

Analytics of Things (AoT) is a term used to describe the analysis of data generated by IoT devices. AoT makes it possible to operate business intelligence within an application rather than a data warehouse. It also assists in understanding patterns, detecting anomalies, predicting potential issues, setting maintenance intervals, and optimizing processes.

Over the years, organizations have relied on data analytics based on a centralized architecture for their data-driven planning and visions of the future. The data is increasing every second, and there is a requirement for a revolutionary approach to overcoming data transfer latency, improving privacy, and meeting customer expectations. The real-time response has especially become necessary after the convergence of AI, 5G, and IoT.

Distributing intelligence across the end of the network provides an opportunity for more efficient data analytics and real-time decision-making at the endpoints with little to no latency. Performing these functions directly on endpoint devices with low-processing capabilities is referred to as endpoint data analytics (**Figure 3**).

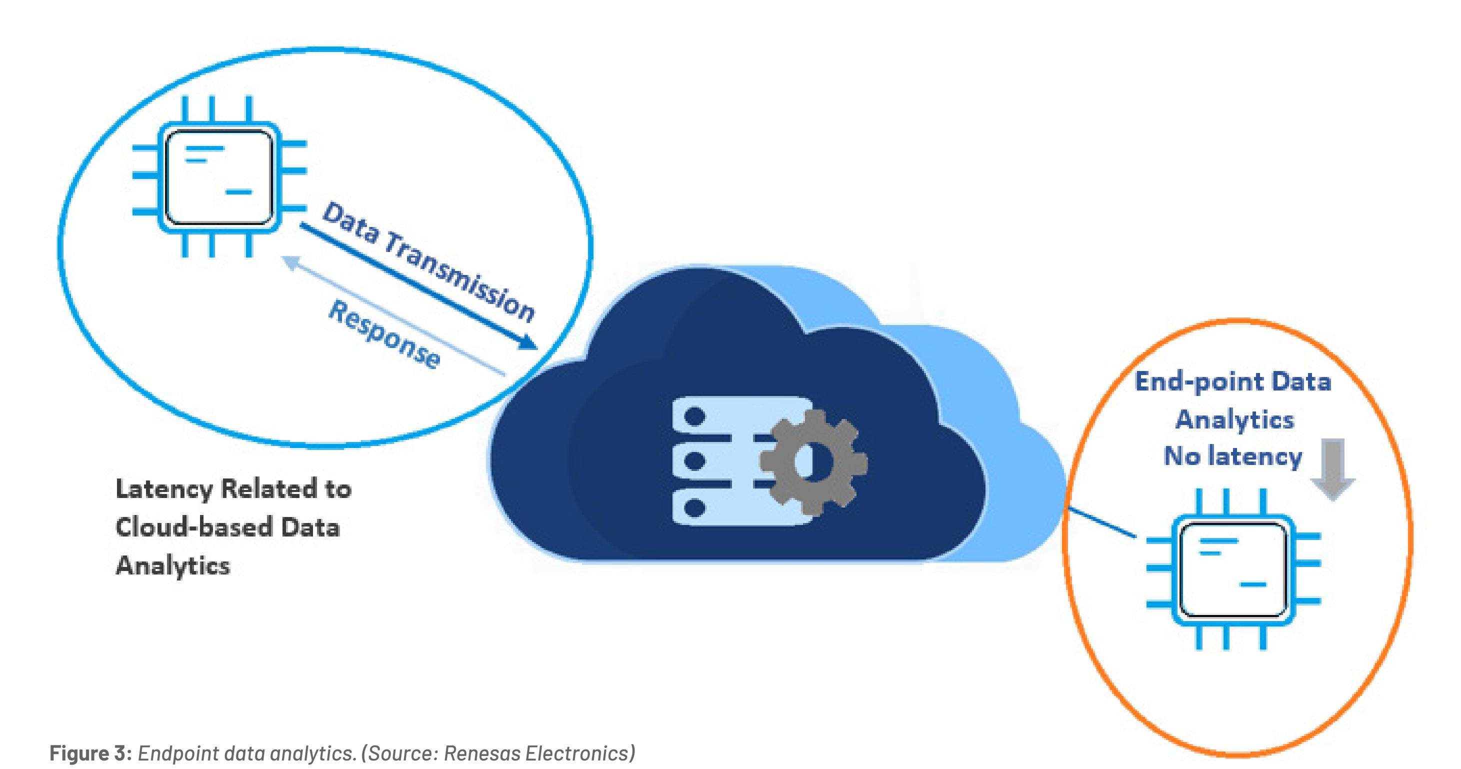


Figure 3: *Endpoint data analytics. (Source: Renesas Electronics)*

The Data Science Support System

Various technologies and sensors facilitate the processing of data analytics at the endpoint. Sensors are required not only to gather and accumulate data but also to assist in assessing and understanding the surrounding environment. Commonly implemented types of sensors include GPS for determining location and thermometers and barometers for measuring temperature and pressure. Cameras, lasers, and radar/sonar can also be used to detect people and/or objects nearby.

Communication technologies such as Wi-Fi and 5G assist in gathering and transmitting data. The data is then fed into data science algorithms that play the central role in assessing the data and recommending potential courses of action. In certain use cases, the outcome of the analytics performed at the

endpoint can even be used to make automatic adjustments to a device's plan of action. An autonomous vehicle, for example, may change routes depending on traffic data collected from other vehicles in the same area or on weather and/or road conditions being monitored by onboard sensors. Data analytics can help categorize potential actions and adjustments into classes to improve the overall decision-making process; for example, the autonomous vehicle can classify potential actions as operational (e.g., optimizing the route for better road and weather conditions) or tactical (e.g., optimizing the route to reduce travel time).

Leveraging the power of predictive maintenance, intelligent sensors embedded in factory equipment can collect data on optimal operations to help manufacturers reduce downtime and avoid potential revenue loss. Anticipating and acting on potential out-of-service issues means that plants can run at peak efficiency.

Where AI and Data Science Go from Here

Traditional business intelligence tools are not structured to process vast amounts of unstructured data. Data science utilizes more advanced tools to assist in sifting, grouping, and analyzing large volumes of data from different sources across multiple interconnected fields. Take marketing, for example: Exploring essential demographic factors like a customer's age, orientation, location, and behavioral patterns can assist in creating highly targeted advertising campaigns. These ads provide more precise recommendations by relying on browsing and purchase history to evaluate the customer's inclination toward buying a specific product. In banking, monitoring unusual customer behavior can aid in detecting and exposing fraud. By scrutinizing and assessing a patient's medical records, data science may help predict and prevent potential medical problems.

Developments in data analytics and AI have transformed the consumer support experience. Real-time analytics respond to the data collected immediately (when its value is at its peak), and data analytics unlock the almost unlimited potential of that data, increasing overall business value and potentially enabling enterprises to secure a high position in the marketplace. Businesses in all sectors rely on machine learning models to make more accurate predictions and effective decisions, thereby improving production efficiency and assisting in overcoming potential security issues in communications between devices and the cloud. Endpoint AI has enhanced the entire process by speeding up the work, providing better results, and improving business decision-making.

Endpoint MCU Implementation of Voice User Interface

GEORGIOS FLAMIS | SENIOR STAFF SYSTEM DESIGN ENGINEER RENESAS ELECTRONICS

Rapid consumer adoption of connected smart speakers and voice assistants suggests that many people prefer to control products with speech rather than counterintuitive button pushes. Consumers also seem to appreciate devices and home appliances that are smart without being connected to the cloud via the internet. The Renesas Electronics voice user interface (VUI) solution, a third-generation reference platform built on the scalable Arm®-based RA2, RA4, and RA6 series of MCUs, is especially targeted to applications in home automation, white goods, and small appliances. The solution gives access to a variety of developer tools and training resources that will take product designs to the next level, including design and training methods of the integrated machine learning model and full evaluation with online and offline methods.

Figure 1: *Daily-use devices with VUI. (Source: Renesas Electronics)*

Introduction

VUI is speech recognition technology that enables users to interact with a computer, smartphone, or other daily-use device with voice commands (**Figure 1**). The unique feature of VUI is the use of voice as primary mode of interaction, in contrast with traditional keyboard, mouse, display, or touch screen.

The new, easy-to-use Renesas hardware platform for VUI solutions is based on the high-performance Renesas Advanced (RA) family of 32-bit microcontroller units (MCUs).

The RA family delivers key advantages compared to competitive Arm Cortex®-M MCUs by providing stronger embedded security, superior CoreMark® performance, and ultra-low-power operation. PSA Certified provides customers the confidence and assurance to quickly deploy secure IoT endpoint and edge devices, and smart factory equipment for Industry 4.0.

The RA family currently includes three product series: RA6, RA4, and RA2. Each of these series has a unique feature set, making them ideal for various applications and market needs. The RA6 Series offers the widest integration of communication interfaces, with integrated Ethernet and TFT display drivers. Flash memory densities range from 256KB to 2MB. The RA6 Series offers up to 240MHz performance running on the Cortex-M4 or Cortex-M33 core with TrustZone®. The RA6 Series supports full security integration, making these devices widely desired for security applications.

The RA4 Series balances the requirements for low power consumption with the demand for connectivity. It offers up to 1MB of flash and a wide range of communication interfaces. The utilized core is the Cortex-M4 or Cortex-M33 with TrustZone and additional security IP integration. Memory densities range from 256KB to 1MB of flash. These devices provide a CPU frequency of up to 100MHz.

The RA2 Series are ideal for designs where the low power requirements of an application matter most. To achieve the best performance, special power-down modes are provided, making these devices well suited for battery-powered applications. The RA2 Series provides memory densities of up to 256KB of embedded flash and a wide single voltage supply range of 1.6 to 5.5V. These devices use the Cortex-M23 core running at up to 48MHz.

The Renesas Flexible Software Package (FSP) is an enhanced software package designed to provide easy-to-use, scalable, high-quality software for embedded system designs using Renesas RA Family microcontrollers (**Figure 2**). It uses an open software ecosystem and provides flexibility in using bare-metal programming, including Azure RTOS, FreeRTOS, other preferred RTOS, legacy code, and third-party ecosystem solutions. The combination of the flexible open architecture of the FSP plus the wide choice of third-party solutions as part of the Arm ecosystem increases the range of choice for application development. This means that developers can choose the software model that best suits their needs while utilizing Renesas's excellent Arm-based silicon solutions as well as speed up the implementation time of complex areas like connectivity and security.

Voice Recognition Engine

Based on the Renesas ecosystem, Cyberon DSpotter (**Figure 3**) is a local voice trigger and command recognition solution with robust noise reduction that consumes very low resources and provides high-accuracy performance. It supports multiple languages as well as many connectivity function and security capable depending on the selected MCU. The major features are listed below:

- Local voice recognition algorithm, no network connectivity needed
- Phoneme-based modeling
- Quick command customization—removes the need to collect speech data in advance
- Optimization by model adaptation just with small amount speech data
- Global language support: 44+ languages worldwide
- Small footprint and cost effective (single DMIC + RA6E1 or RA4E1)
- DSMT Tool: wake-word and commands customization, performance tuning, testing, no prior neural network knowledge needed.
- Separation of recognition engine and command model, switching commands dynamically
- Low-power, high-efficiency RA MCU with strong security function

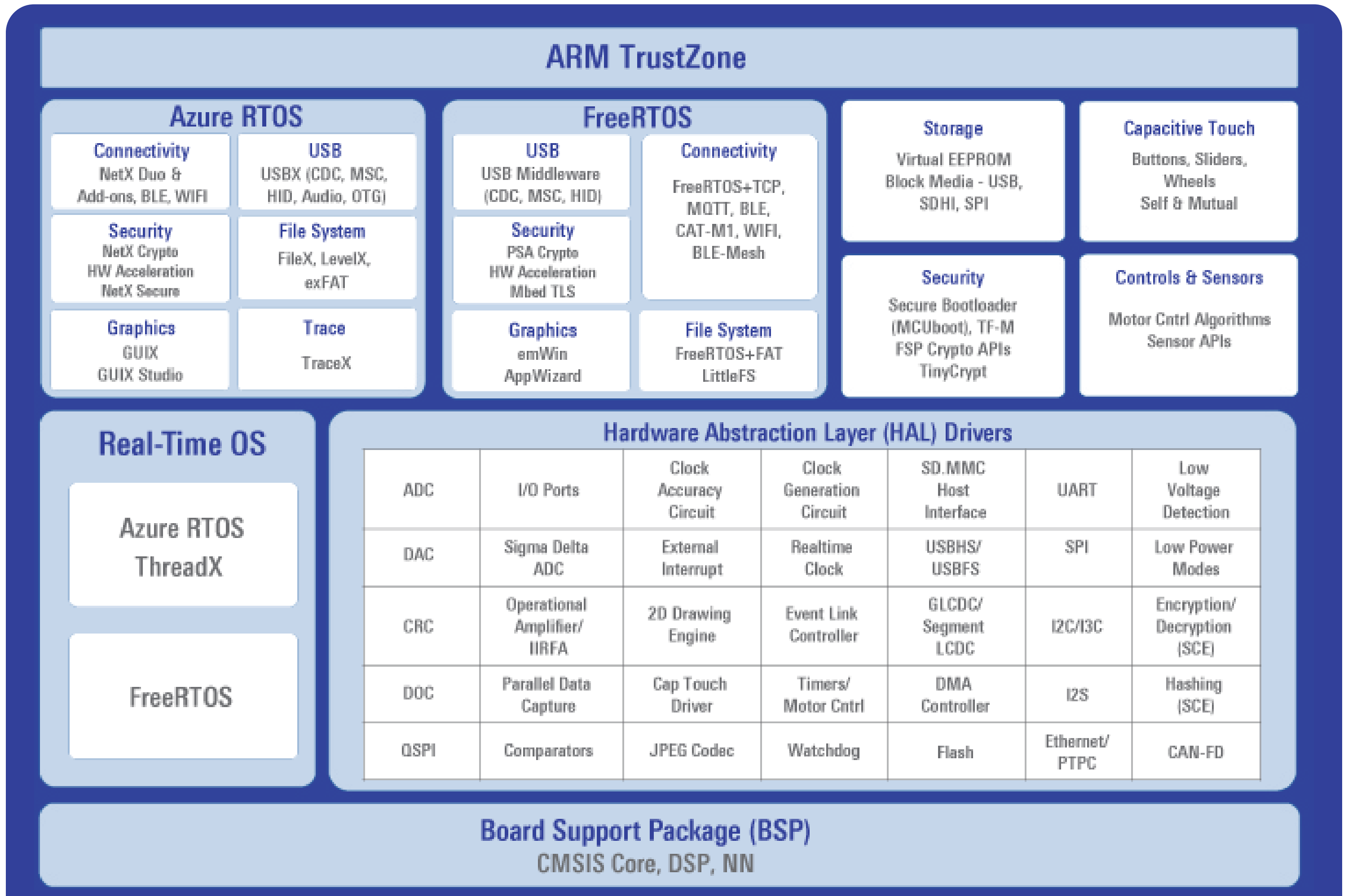


Figure 2: *Renesas Flexible Software Package. (Source: Renesas Electronics)*

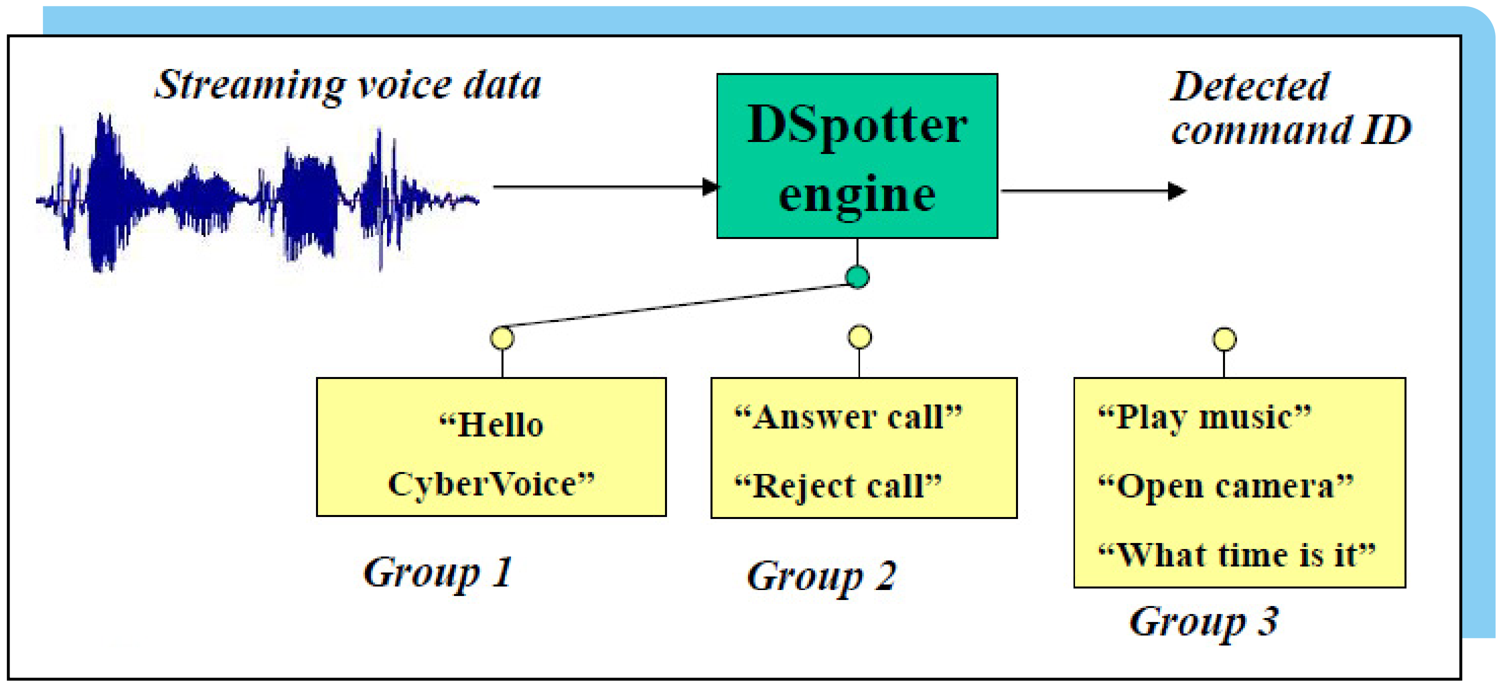


Figure 3: *Cyberon DSpotter voice recognition engine. (Source: Renesas Electronics)*

Table 1: *Results of hit rate*

SNR	Background noise	Distance	Hit-Rate	Alexa Requirements
(Clean)	none	1.5m	**100.00%**	90%
(Clean)	none	3m	**100.00%**	90%
10dB	Babble	1.5m	**98.55%**	80%
10dB	Babble	3m	**98.84%**	80%
10dB	Music	1.5m	**98.26%**	80%
10dB	Music	3m	**98.55%**	80%
10dB	TV	1.5m	**98.84%**	80%
10dB	TV	3m	**98.55%**	80%
5dB	Babble	1.5m	**98.84%**	80%
5dB	Babble	3m	**96.24%**	80%
5dB	Music	1.5m	**98.84%**	80%
5dB	Music	3m	**97.08%**	80%
5dB	TV	1.5m	**93.37%**	80%
5dB	TV	3m	**90.72%**	80%

Results

Hit rate has been captured with voice commands mixed with different types of noise in levels suitable to create distinguished signal-to-noise ratios. The test bench shown in **Figure 4** is utilized and the results are summarized in **Table 1**.

Conclusions

The complete implementation of an endpoint voice commands recognition system has been presented that is capable to execute on a simple MCU device. The reference design enables local voice recognition without a network connection and allows to quickly start building an enhanced VUI in minutes that recognizes voice commands to trigger the corresponding operation.

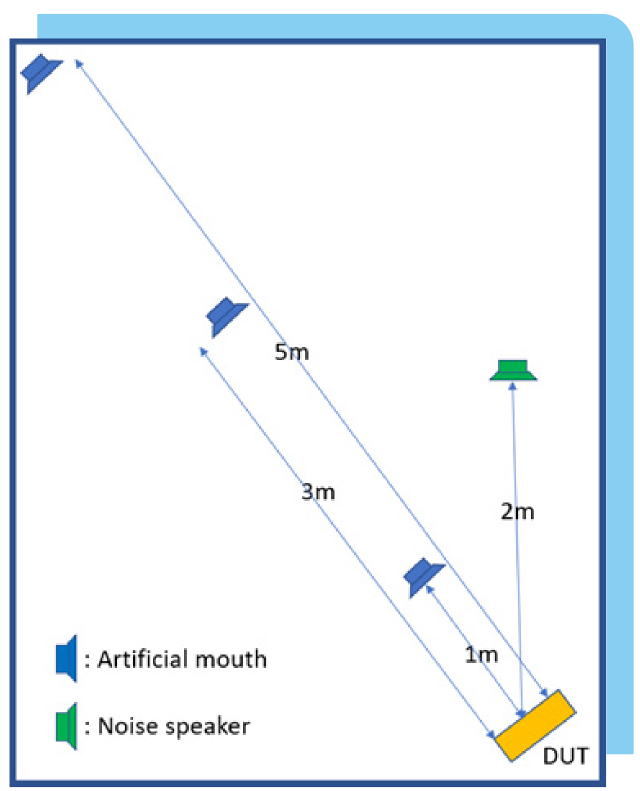


Figure 4: *Evaluation test bench. (Source: Renesas Electronics)*

Reality AI: A Game Changer in Sensor Data Analysis

JOSEPH DOWNING | MOUSER ELECTRONICS

Unscheduled equipment downtime can be costly for companies—both financially and in time spent on repairs. By having systems to monitor and report on potential equipment issues, companies can catch potential issues before they become major problems. This helps them take corrective action quickly and avoid losing resources due to unscheduled or extended equipment downtime. It also helps them identify issues that can be quickly and easily fixed, saving time and money in the long run.

To mitigate downtime, predictive maintenance using machine learning (ML) algorithms can be trained and deployed based on data gathered from sensor information, providing early detection in critical systems. Renesas Reality AI has revolutionized the way businesses analyze sensor data. The platform uses ML algorithms and advanced analytics to extract meaningful insights from data streams generated by equipment sensors. With its powerful tools, Reality AI has become a game changer in the field of sensor data analysis.

What Is Reality AI?

Renesas Reality AI is a software tool that allows developers to develop intelligent, embedded artificial intelligence (AI) applications quickly and easily for edge devices. The suite of tools includes an intuitive drag-and-drop interface for data upload, pre-trained ML models, and automatic code generation for Renesas's e^2 studio integrated development environment (IDE), allowing developers to build and deploy applications faster and with less effort.

> "Reality AI has become a game changer in the field of sensor data analysis."

Renesas Reality AI supports a wide range of use cases, including predictive maintenance, anomaly detection, and voice. Using its robust ML capabilities, developers can easily train and optimize models using their own data and quickly deploy those models to their target devices. The tool also includes comprehensive support for a wide range of Renesas hardware platforms and microcontrollers (MCUs), making it easy to integrate with existing hardware and software systems.

Renesas Reality AI supplies a powerful and efficient solution for developers looking to build intelligent, embedded AI applications for edge devices.

Features of Reality AI

The Reality AI platform offers several key features that make it a valuable resource for businesses looking to analyze sensor data:

- Anomaly detection: By identifying anomalies in the data, the platform can automatically generate ML models, explanatory visualizations, and hardware design analytics. This can be especially useful for predictive maintenance applications, where identifying issues early can prevent costly downtime and repairs.

- Advanced signal processing: Reality AI automatically searches a wide range of signal-processing transforms to create a custom, optimized feature transform.
- MCU/MPU edge nodes: Reality AI runs on almost every MCU and microprocessor (MPU) core available from Renesas, with new ones added constantly. Reality AI also supports Renesas Motor Control boards.
- Data visualization: The platform offers a user-friendly interface that allows businesses to visualize their data in real time. This can be useful for monitoring operations and identifying trends or issues as they occur.

Applications of Reality AI

The Reality AI platform has a wide range of applications in numerous industries, including industrial automation; heating, ventilation, and air conditioning (HVAC); and automotive. The following are just a few examples of how the platform could be used:

- Industrial anomaly detection: Through a hardware and software solution for equipment monitoring and end-of-line testing for quality control, users can deploy an edge node with one or more included sensors and begin monitoring production equipment for unusual or anomalous functions.
- HVAC and refrigeration systems: By combining ML and edge processing and using data such as vibration and electrical signals, this application can potentially predict a wide array of anomalies, faults, and operating conditions.
- Automotive sound recognition: By analyzing audio data, the solution can detect and localize potential hazards (e.g., emergency vehicles, motorcycles) that may be hidden from view from line-of-sight sensors.
- Predictive maintenance: By analyzing sensor data from industrial machines, the platform can identify potential issues before they occur. This can prevent costly downtime and repairs and increase the lifespan of the machines.

Sensorless Motor Demo

With support provided for the Renesas Motor Control Kits, one way to evaluate the features available through Reality AI Tools is by using an existing demo, such as the Reality AI demo on mouser.com. The demo incorporates the Renesas RA6T2 Motor Control Kit and provides insight and instructions for creating a predictive sensorless ML model that detects unbalanced motor loads using current, voltage, and speed. The RA6T2 Motor Control Kit features an enhanced central processing unit (CPU) and hardware accelerator for home appliance and industrial automation application deployments. The kit comes complete with all the basics, including a CPU board, inverter board, communication board, and brushless DC motor.

The project article outlines the necessary resources and required hardware, with applicable links as well as steps to train and deploy your ML model. If you are looking to jump right in with Reality AI without the hardware, pre-built data files are also available to help you get started. The steps in the demo take you through everything from uploading and curating your data to training and deployment. Of course, this only scratches the surface of the entire suite of available features, such as using new data to further test and optimize as well as ways to potentially decrease the complexity of the model without sacrificing accuracy.

An example of a simple application for this type of demo is detecting issues with load balance in a washing machine during a spin cycle and notifying the end user.

Conclusion

Renesas Reality AI is a powerful software toolset that can revolutionize the way businesses analyze sensor data. Through automated analysis and selection of the most relevant sensor data for the specific prediction model of interest, the Reality AI toolset extracts meaningful insights and offers potential BOM optimization. These capabilities can also be incorporated via firmware updates into existing systems without additional BOM costs. The models generated are extremely resource efficient, consuming insignificant CPU load so the MCU is practically free to perform all the usual tasks while incorporating these features. With its complete integration into the Renesas toolchain and full support for the MCU portfolio and development boards, the Reality AI toolset provides the speed and accuracy required to deliver the precision businesses expect to make sense of vast amounts of data, helping to minimize costly equipment downtime.