Embedded Sensor Platform with AI Algorithms—Locally from Big Data to Smart Data

Dzianis Lukashevich, Director of Platforms and Solutions

Industry 4.0 applications generate a huge volume of complex data—big data. The increasing number of sensors and, in general, available data sources require the virtual view of machines, systems, and processes to be ever more detailed. This naturally increases the potential for generating added value along the entire value chain. At the same time, however, the question as to how exactly this potential can be extracted keeps arising—after all, the systems and architectures for data processing are becoming more and more complex, and the number of sensors and actuators is constantly increasing. Only with relevant, high quality, and useful data—smart data—can the associated economic potential be unfolded.

Challenges

Collecting all possible data and storing them in the cloud in the hopes that they will later be evaluated, analyzed, and structured is still a widespread, but not particularly effective, approach. The potential for generating added value from the data remains unused; finding a solution later on becomes more complex and costly. A better alternative is to make conceptual considerations early on to determine which information is relevant to the application and where in the data flow the information can be extracted (see Figure 1). Figuratively speaking, this means refining the data—for example, making smart data out of big data for the entire processing chain. At the application level, a decision can already be made regarding which AI algorithms have a high probability of success for the individual processing steps. This depends on boundary conditions such as available data, application type, available sensor modalities, and background information about the lower level physical processes.

For the individual processing steps, correct handling and interpretation of the data are extremely important for real added value to be generated from the sensor signals. Depending on the application, it may be difficult to interpret the discrete sensor data correctly and extract the desired information. Often the temporal behavior plays a role and has a direct effect on the desired information. In addition, the dependencies between multiple sensors must frequently be taken into account. For complex tasks, simple threshold values and manually determined logic are no longer sufficient or do not allow for automated adaptation to changing environmental conditions.

Figure 1. Division of the algorithm pipeline into embedded, edge, and cloud platforms. Note: this article focuses on embedded platforms.
Embedded, Edge, or Cloud AI Implementation?

The overall data processing chain with all the algorithms needed in each individual step must be implemented in such a way that the highest possible added value can be generated. Implementation usually occurs at all levels—from the small sensor with limited computing resources through gateways and edge computers to large cloud computers. It is clear here that the algorithms should not only be implemented at one level. Rather, in most cases, it is more advantageous to implement the algorithms as close as possible to the sensor. Through this, the data are compressed and refined at an early stage, and the communication and storage costs are reduced. In addition, through early extraction of the essential information from the data, development of global algorithms at the higher levels is less complex. In most cases, algorithms from the streaming analytics area are also useful for avoiding unnecessary storage of data and, thus, high data transfer and storage costs. These algorithms use each data point only once; for example, the complete information is extracted directly, and the data do not need to be stored.

Embedded Platform for Condition-Based Monitoring

The ARM® Cortex®-M4F processor-based open embedded platform iCOMOX from Shiratech Solutions, Arrow, and Analog Devices is an extremely power-saving, integrated microcontroller system with integrated power management, as well as analog and digital sensors and peripheral devices for data acquisition, processing, control, and connectivity. All of this makes it a very good candidate for local data processing and early refinement of data with state-of-the-art smart AI algorithms.

iCOMOX stands for intelligent condition monitoring box, and it can be used for entry into the world of structural health and machine condition monitoring based on vibration, magnetic fields, sound, and temperature analysis. On request, the platform can be supplemented with additional sensor modalities—for example, gyroscopes from Analog Devices for precise measurement of rotational speeds, even in environments with high shock and vibration loads (see Figure 2). The AI methods implemented in the iCOMOX can deliver a better estimate of the current situation through so-called multisensor data fusion. In this way, various operating and fault conditions can be classified with better granularity and higher probability. Through smart signal processing in the iCOMOX, big data becomes smart data, making it necessary for only the data relevant to the application case to be sent to the edge or the cloud.

For wireless communications, the iCOMOX provides a solution with high reliability and robustness as well as extremely low power consumption. The SmartMesh® IP network is composed of a highly scalable, self-forming/optimizing multihop mesh of wireless nodes that collect and relay data. A network manager monitors and manages the network performance and security and exchanges data with a host application. The intelligent routing of the SmartMesh IP network determines an optimum path for each individual packet in consideration of the connection quality, the schedule for each packet transaction, and the number of multihops in the communication link.

Especially for wireless, battery-operated condition monitoring systems, embedded AI can help extract the full added value. Local conversion of sensor data to smart data by the AI algorithms embedded in the iCOMOX results in a lower data flow and consequently less power consumption than is the case with direct transmission of raw sensor data to the edge or the cloud.

Range of Applications

The iCOMOX, including the AI algorithms developed for it, has a wide range of applications in the field of monitoring machines, systems, structures, and processes—extending from detection of anomalies to complex fault diagnostics and immediate initiation of fault elimination. Through the integrated microphone and accelerometer, magnetic field sensor, and temperature sensor, the iCOMOX enables, for example, monitoring of vibrations and noises, as well as other operating conditions in diverse industrial machines and systems. Process states, bearing or rotor and stator damage, failure of the control electronics, etc., and even unknown changes in system behavior, for example, due to damage to the electronics, can be detected by AI. If behavior models are available for certain damages, these damages can even be predicted. Through this, maintenance measures can be taken at an early stage and, thus, unnecessary damage-based failure can be avoided. If no predictive model exists, the embedded platform can also help subject matter experts successively learn the behavior of a machine and over time derive a comprehensive model of the machine for predictive maintenance. In addition, the iCOMOX can be used to optimize the complex manufacturing processes to achieve a higher yield or better product quality.

Embedded AI Algorithms for Smart Sensors

With data processing by AI algorithms, automated analysis is even possible for complex sensor data. Through this, the desired information and, thus, added value are automatically arrived at from the data along the data processing chain. Selection of an algorithm often depends on existing knowledge about the application. If extensive domain knowledge is available, AI plays a more supporting role and the algorithms used are quite rudimentary. If no expert knowledge exists, the algorithms can be much more complex. In many cases, it is the application that defines the hardware and, through this, the limitations for the algorithms.

For the model building, which is always a part of an AI algorithm, there are basically two different approaches: data-driven approaches and model-based approaches.
Anomaly Detection Using Data-Driven Approaches

If only data, but no background information that could be described in the form of mathematical equations, are available, then so-called data-driven approaches must be chosen. These algorithms extract the desired information (smart data) directly from the sensor data (big data). They encompass the full range of machine learning methods, including linear regression, neural networks, random forest, and hidden Markov models.

A typical algorithm pipeline for data-driven approaches that can be implemented on embedded platforms such as the iCOMOX are composed of three components (see Figure 3): 1) data preprocessing, 2) feature extraction and dimensionality reduction, and 3) the actual machine learning algorithm.

During data preprocessing, the data are processed in such a way that the downstream algorithms, especially the machine learning algorithms, converge to an optimum solution within the shortest possible computational time. Missing data must thereby be replaced using simple interpolation methods in consideration of the time dependence and the interdependence between different sensor data. Furthermore, the data are modified by prewhitening algorithms in such a way that they appear to be mutually independent. As a result of this, there are no more linear dependencies in time series or between sensors. Principal component analysis (PCA), independent component analysis (ICA), and so-called whitening filters are typical algorithms for prewhitening.

During feature extraction, characteristics, also known as features, are derived from the preprocessed data. This part of the processing chain strongly depends on the actual application. Due to the limited computing power of embedded platforms, it is not yet possible here to implement computationally intensive, fully automated algorithms that evaluate the various features and use specific optimization criteria to find the best features—genetic algorithms would be included among these. Rather, for embedded platforms such as the iCOMOX that have low power consumption, the method used for extracting features must be specified manually for each individual application. The possible methods include transforming the data into the frequency domain (fast Fourier transformation), applying a logarithm to the raw sensor data, normalizing the accelerometer or gyroscope data, finding the largest eigenvectors in PCA, or performing other calculations on the raw sensor data. Different algorithms for feature extraction can also be selected for different sensors. A large feature vector containing all the relevant features from all of the sensors is obtained as a result.

If the dimensionality of this vector exceeds a certain size, it must be reduced through dimensionality reduction algorithms. The minimum and/or maximum values within a certain window can simply be taken, or more complex algorithms such as the previously mentioned PCA or self-organizing maps (SOM) can be used for this purpose.

Only after the complete preprocessing of the data and the extraction of the features relevant to the respective application can the machine learning algorithms be optimally employed to extract different information right on the embedded platform. As was the case for feature extraction, the selection of the machine learning algorithm strongly depends on the respective concrete application. Fully automated selection of the optimum learning algorithm—for example, via genetic algorithms—is also not possible due to the limited computing power. However, even somewhat more complex neural networks, including the training phase, can be implemented on embedded platforms such as the iCOMOX. The decisive factor here is the limited available memory. For this reason, the machine learning algorithms, as well as all previously mentioned algorithms in the entire algorithm pipeline, must be modified in such a way that the sensor data are directly processed. Each data point is used only once by the algorithms; for example, all of the relevant information is extracted directly, and the memory-intensive collection of large amounts of data and the associated high data transfer and storage costs are eliminated. This type of processing is also known as streaming analytics.

![Algorithm Pipeline on Embedded Platform—Data-Driven Approach](image)

Figure 3. Data-driven approaches for embedded platforms.
The previously mentioned algorithm pipeline was implemented on the iCOMOX and evaluated for anomaly detection in two different applications: condition-based monitoring for ac motors and trajectory monitoring in industrial robots. The algorithms were basically the same for both applications; only the parameterization differed in that the time interval under consideration was short for motor monitoring and long for trajectory monitoring. Through limitation of the hardware, different values were also derived for the remaining algorithm parameters. The accelerometer and gyroscope data with a sampling rate of 1 kHz each were used as input data. For the motor condition monitoring, the microphone data were also used as input data so as to include the acoustic peculiarities and thereby improve the anomaly detection accuracy. The results of the local calculation on the embedded platform are shown in Figure 4 and Figure 5. In both examples, the accelerometer and gyroscope data, the locally derived features, and the locally calculated anomaly indicator are presented. This indicator increases sharply with new signal behavior and is much lower on reoccurrence; that is, the newly detected signal was considered and updated in the model by the learning algorithm.

Dynamic Pose Estimation Using Model-Based Approaches

Another fundamentally different approach is modeling by means of formulas and explicit relationships between the sensor data and the desired information. These approaches require the availability of physical background information or system behavior in the form of a mathematical description. These so-called model-based approaches combine the sensor data with this background information to yield a more precise result for the desired information. Some of the best known examples here are the Kalman filter (KF) for linear systems and the unscented Kalman filter (UKF), the extended Kalman filter (EKF), and particle filter (PF) for nonlinear systems. The selection of the filter strongly depends on the respective application.

A typical algorithm pipeline for model-based approaches that can be implemented on embedded platforms such as the iCOMOX are composed of three components (see Figure 6): 1) outlier detection, 2) prediction step, and 3) filtering step.
During outlier detection, sensor data far removed from the actual estimate of the system condition are either fractionally weighted or taken out completely in further processing. Through this, more robust data processing is achieved.

In the prediction step, the current system condition is updated over time. This is done with the help of a probabilistic system model that describes a prediction of the future system condition. This probabilistic system model is often derived from a deterministic system equation that describes the dependence of the future system condition on the current system condition as well as other input parameters and disturbances. In the example of condition monitoring in an industrial robot considered here, this would be the dynamic equation for the individual articulated arms, which only allow certain directions of motion at any point in time.

In the filtering step, the predicted system condition is then processed with a given measurement and the condition estimate thereby updated. There is a measurement equation equivalent to the system equation that enables the relationship between the system condition and the measurement to be described in a formula. For the position estimation considered here, this would be the relationship between the accelerometer and gyroscope data and the precise position of the sensor in space.

Combination of the data-driven and model-based approaches is both conceivable and advantageous for certain applications. The parameters of the underlying models for the model-based approaches can, for example, be determined through the data-driven approaches or dynamically adapted to the respective environmental condition. In addition, the system condition from the model-based approach can be part of a feature vector for the data-driven approaches. However, all of this strongly depends on the respective application.

The previously mentioned algorithm pipeline was implemented on the iCOMOX and evaluated for precise dynamic pose estimation in an industrial robot end effector. Accelerometer and gyroscope data with a sampling rate of 200 Hz each were used as input data. The iCOMOX was attached to the end effector of the industrial robot and its pose—consisting of position and orientation—was determined. The results are shown in Figure 7. As shown, the direct calculation leads to very fast reactions, but also to a large amount of noise with numerous outliers. An IIR filter, as is commonly used in practice, leads to a very smooth signal, but it follows the true pose very poorly. In contrast, the algorithms presented here lead to a very smooth signal where the estimated pose very precisely and dynamically follows the motion of the end effector of the industrial robot.

![Algorithm Pipeline on Embedded Platform—Model-Based Approach](image)

**Figure 6.** Model-based approaches for embedded platforms.

![Precise Dynamic Angle Estimation](image)

**Figure 7.** Precise dynamic angle estimation on an embedded platform. The implemented algorithm showed much better performance when compared to the direct calculation and IIR filtering.
Conclusion

Ideally, through the corresponding local data analysis, the AI algorithms should also be able to decide themselves which sensors are relevant for the respective application and which algorithm is the best one for it. This means smart scalability of the platform. At present it is still the subject matter expert who must find the best algorithm for the respective application, even though the AI algorithms used here can already be scaled with minimal implementation effort for various applications for machine condition and structural health monitoring.

The embedded AI should also make a decision regarding the quality of the data and, if it is inadequate, find and make the optimal settings for the sensors and the entire signal processing. If several different sensor modalities are used for the fusion, the weaknesses and disadvantages of certain sensors and methods can be compensated for by the use of an AI algorithm. Through this, the data quality and the system reliability are increased. If a sensor is classified as not or not very relevant to the respective application by the AI algorithm, its data flow can be accordingly throttled.

The open embedded platform iCOMOX from Shiratech Solutions, Arrow, and Analog Devices is available through Arrow and contains a free software development kit and numerous example projects for hardware and software for accelerating prototype creation, facilitating development, and realizing original ideas. A robust and reliable wireless mesh network of smart sensors for condition-based monitoring can be created using multisensor data fusion and embedded AI. With it, big data is locally turned into smart data.

References

iCOMOX Data Sheet, Shiratech Solutions.

About the Author

Dzianis Lukashevich is the director of platforms and solutions at Analog Devices. His focus is on megatrends, emerging technologies, complete solutions, and new business models shaping the future of industries and transforming ADI business in the broad market. Dzianis joined ADI Sales and Marketing in Munich, Germany, in 2012. He received his Ph.D. in electrical engineering from Munich University of Technology in 2005 and M.B.A. degree from Warwick Business School in 2016. He can be reached at dzianis.lukashevich@analog.com.