

Overview of AI-Models and Tools in Embedded IIoT Applications

Pierpaolo Dini ^{1,*} , Lorenzo Diana ² , Abdussalam Elhanashi ¹  and Sergio Saponara ¹ 

¹ Department of Information Engineering, University of Pisa, Via G. Caruso 16, 56100 Pisa, Italy; abdussalam.elhanashi@ing.unipi.it (A.E.); sergio.saponara@unipi.it (S.S.)

² Independent Researcher, 56100 Pisa, Italy; ldiana.res@libero.it

* Correspondence: pierpaolo.dini@ing.unipi.it

Abstract: The integration of Artificial Intelligence (AI) models in Industrial Internet of Things (IIoT) systems has emerged as a pivotal area of research, offering unprecedented opportunities for optimizing industrial processes and enhancing operational efficiency. This article presents a comprehensive review of state-of-the-art AI models applied in IIoT contexts, with a focus on their utilization for fault prediction, process optimization, predictive maintenance, product quality control, cybersecurity, and machine control. Additionally, we examine the software and hardware tools available for integrating AI models into embedded platforms, encompassing solutions such as Vitis AI v3.5, TensorFlow Lite Micro v2.14, STM32Cube.AI v9.0, and others, along with their supported high-level frameworks and hardware devices. By delving into both AI model applications and the tools facilitating their deployment on low-power devices, this review provides a holistic understanding of AI-enabled IIoT systems and their practical implications in industrial settings.

Keywords: artificial intelligence; machine learning; deep learning; computer vision; industrial internet of things; embedded systems; embedded devices; SW/HW tools for AI; embedded ML; TinyML; embedded low-power



Citation: Dini, P.; Diana, L.; Elhanashi, A.; Saponara, S. Overview of AI-Models and Tools in Embedded IIoT Applications. *Electronics* **2024**, *13*, 2322. <https://doi.org/10.3390/electronics13122322>

Academic Editor: Baris Aksanli

Received: 14 May 2024

Revised: 6 June 2024

Accepted: 9 June 2024

Published: 13 June 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introductions

1.1. Motivations and Contributions

In recent years, the Industrial Internet of Things (IIoT) has revolutionized the industrial sector, enabling the creation of intelligent interconnected systems capable of improving operational efficiency, predictive maintenance, and product quality. However, the true value of the IIoT can only be fully realized by integrating Artificial Intelligence (AI) models into embedded platforms, including microprocessors and microcontrollers, which underpin the embedded devices and systems commonly used in the industrial environment. The following reasons highlight the importance and urgency of this integration:

- **Reduced latency and communication overhead:** Integrating AI models directly into embedded devices allows data to be processed and analyzed on-site, reducing the need to transmit large amounts of data to remote servers for processing. This significantly reduces latency and communication overhead, enabling faster and more responsive decisions [1–3].
- **Energy savings and resource optimization:** Processing data directly on embedded platforms reduces overall system power consumption, as it eliminates the need to transmit data over long distances and run complex artificial intelligence algorithms on remote servers. Furthermore, optimizing the computing and memory resources of embedded devices allows for efficient implementation of artificial intelligence models even in the presence of resource constraints [4–7].
- **Improved data security and privacy:** Integrating AI models on embedded platforms helps to keep sensitive and critical data within the enterprise perimeter, reducing the risk of security breaches and improving data privacy. Furthermore, local data

processing allows encryption and cybersecurity techniques to be applied directly on embedded devices, ensuring greater protection of sensitive information [8–11].

- **Increased system resilience and availability:** The integration of artificial intelligence models on embedded platforms makes IIoT systems more resilient and autonomous, capable of continuing to operate even in the absence of a network connection or in adverse environmental conditions. This is especially crucial in industrial environments where business continuity is essential for the safety and efficiency of operations [12–14].

In summary, the integration of artificial intelligence models on embedded platforms represents a fundamental step towards unlocking the full potential of the Industrial Internet of Things, enabling the implementation of intelligent, efficient, and secure systems directly into the fabric of industrial infrastructure. This rationale section provides an overview of the main reasons that such integration is essential to the success of the IIoT and the industrial sector as a whole. In the panorama of the current literature on Artificial Intelligence (AI) applied to embedded systems in the Industrial Internet of Things (IIoT), many works mainly focus on the presentation and analysis of different AI models and their specific applications. However, this article stands out for offering a comprehensive and in-depth review that goes beyond merely describing AI models. Rather, it aims to critically examine the various tools, frameworks and methodologies used for the effective integration of such models on embedded platforms. While existing studies can offer a broad overview of available AI models, this review goes further by exploring the challenges, solutions, and best practices related to integrating such models into the specific domain of IIoT. Through an in-depth synthesis of academic sources, industry publications, and practical experiences, this article aims to provide readers with a comprehensive and informative guide to understanding and addressing the complexities of integrating AI on embedded systems in the IIoT.

1.2. Background on the IIoT

The Industrial Internet of Things (IIoT) refers to the connection between physical devices and the digital world via the internet, enabling these devices to collect and exchange data. Such devices can be virtually anything, from simple sensors to complex industrial machines, wearables, household appliances, and much more. The main goal of the IIoT is to create an interconnected ecosystem capable of enhancing efficiency, productivity, and user experience (see Figure 1). **Peripheral Devices** or edge devices form the physical basis of the Internet of Things (IoT), collecting data directly from the real world. These can include a wide range of devices, such as sensors, actuators, cameras, smartwatches, smart meters, and more. They are the first point of contact for data acquisition, and represent the point of origin of information in the IIoT ecosystem. Their large-scale deployment allows for the collection of data from multiple sources, enabling contextual analysis and action [15,16]. In many cases, edge devices do not have the ability to connect directly to the internet or back-end network; therefore, **Gateways** are used to collect data from multiple devices, aggregate it, and send it to a centralized infrastructure via appropriate communication protocols such as MQTT, CoAP, or HTTP. Gateways act as a bridge between edge devices and the communications infrastructure, facilitating efficient data transfer and enabling the connection of heterogeneous devices [17,18]. The data collected by the devices is transferred to a **Cloud or Fog Computing** infrastructure for processing, analysis and storage. The cloud offers virtually unlimited computing resources and scalability, enabling complex analytics and long-term data storage. On the other hand, fog computing moves some of the data processing closer to the devices to reduce latency and bandwidth usage. This approach is particularly useful in scenarios where data timeliness is critical, such as in manufacturing or video surveillance [19–21].

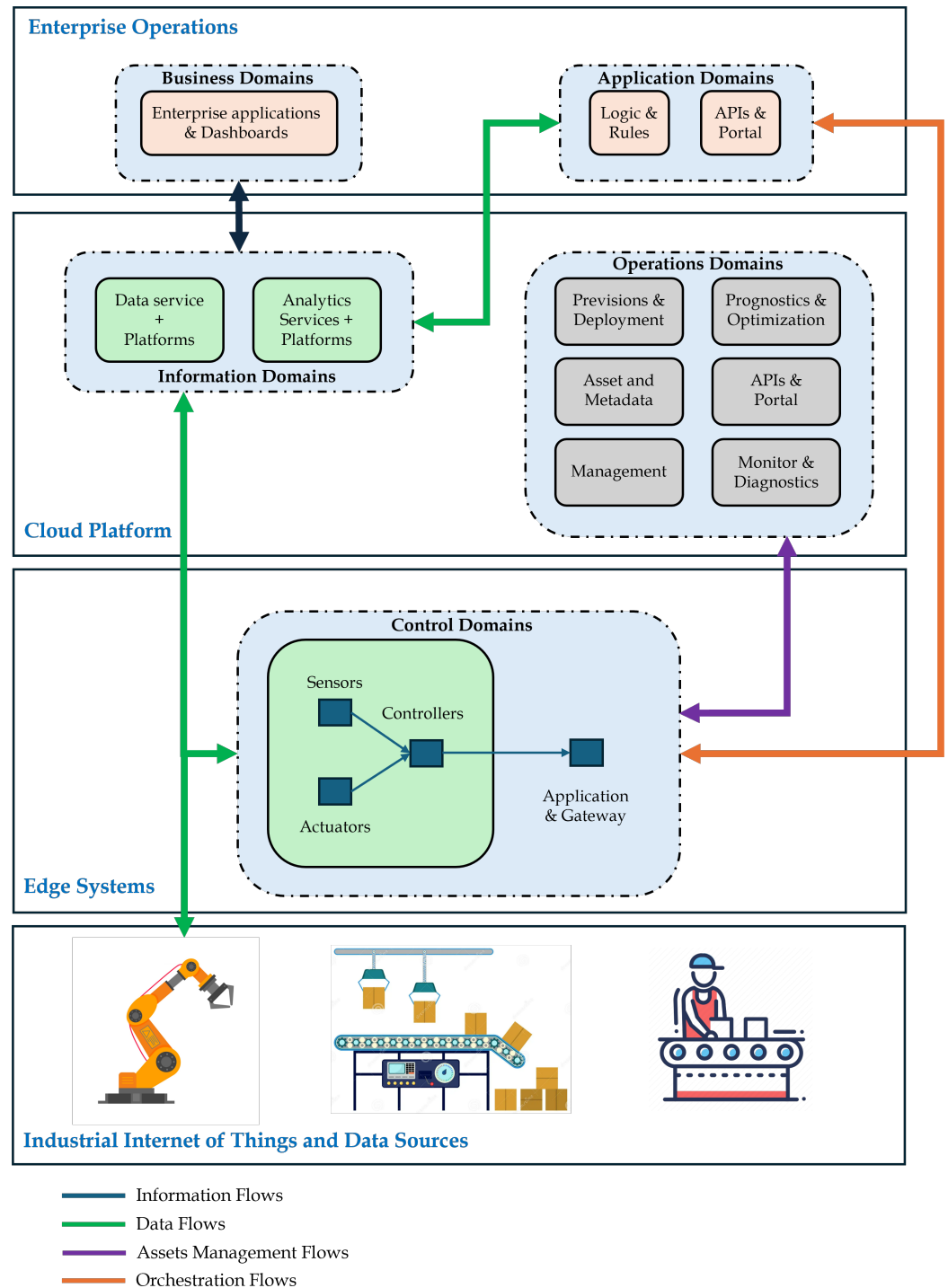


Figure 1. Schematic representation of IIoT systems architecture.

The **IoT Platform** is the software component that manages the entire IIoT infrastructure. It includes features such as device management, data analytics, security, and APIs for IIoT application development. The platform plays a critical role in orchestrating operations within the IIoT ecosystem, enabling centralized device management, data collection and analysis, and delivery of services and applications [22,23]. **IoT Applications and Services** are what end users directly interact with. They can include monitoring dashboards, mobile apps, remote control systems, home and industrial automation solutions, and so on. These tools allow users to view, manage, and interact with IIoT devices and data intuitively and effectively (see Figure 2).

Applications and services represent the end point of the IIoT ecosystem, providing tangible value to end users and facilitating the adoption and use of IIoT technology [24–27]. **Sensors** are critical devices in the IIoT ecosystem; they detect and measure crucial environmental data such as temperature, humidity, light intensity, motion, pressure, and more. These data is critical to understanding the surroundings and fueling decision-making within the IIoT environment. For example, temperature sensors can monitor conditions in a warehouse to ensure that heat-sensitive products are stored properly, while motion sensors can activate security systems in the event of an intrusion [28–33]. **Actuators** are responsible for acting on the physical world in response to commands received from an IIoT system. These devices can perform a wide range of actions, such as turning on a light, activating a pump, adjusting a valve, and more. Actuators play a crucial role in automating processes and implementing corrective or preventive actions based on data collected by sensors. For example, an actuator can be used to automatically close a valve if a liquid leak from a tank is detected [34–36]. **Wearable Devices** such as smartwatches, fitness bands and smart glasses have become increasingly popular in the IIoT space. These devices track physical activities, health parameters, location, and more, offering users a wide range of useful features for monitoring and improving personal wellbeing. For example, a smartwatch can constantly monitor an individual's heartbeat and alert them if there are any abnormalities [37,38]. **Smart Home Devices** such as smart thermostats, smart light bulbs, and security cameras enable remote control and monitoring of home environments. These devices offer users greater comfort, safety, and energy efficiency, allowing them to adjust the temperature, turn lights on and off, and monitor home activities remotely via dedicated apps [39–42]. **Industrial Devices** are designed specifically for industrial environments, and can include production monitoring sensors, computer numerical control (CNC) machines, autonomous industrial vehicles, and more. These devices are essential for optimizing production processes, improving efficiency, and ensuring safety in the workplace. For example, monitoring sensors can collect real-time production data to identify inefficiencies and improve overall plant performance. **Wired Communication Systems** use protocols such as ethernet or modbus over physical cables to transmit data between IIoT devices and communication infrastructures. These systems offer a stable and reliable connection that is ideal for applications requiring high-speed and low-latency data transfers [43–46]. **Wireless Communication Systems** such as WiFi, Bluetooth, Zigbee, LoRa, NB-IoT, and Sigfox offer flexibility and are easier to install than wired options. These protocols allow data transmission over long distances without the need for physical cables, making them ideal for applications where mobility or flexible deployment of devices is important [47–49]. **Hybrid Communication Systems** combine wired and wireless connections to ensure connectivity in heterogeneous environments. This approach allows users to take advantage of the advantages of wired connections in terms of stability and reliability together with the flexibility and ease of installation of wireless connections. This solution is best suited for scenarios where it is necessary to balance performance and flexibility requirements [50–53].

Security represents a fundamental element in the Internet of Things (IoT), aiming to protect sensitive data and prevent malicious attacks that could compromise the integrity, confidentiality, and availability of information (see Figure 3). Security measures in the IIoT are multidimensional and involve several key aspects, ranging from protecting the devices themselves to managing data transmission and storage [54–62]. One of the first lines of defense is device authentication, which ensures that only authorized devices can access network resources. This can be achieved through the implementation of robust authentication protocols and the adoption of appropriate authorization policies. **Data Encryption** is another crucial practice for protecting sensitive information during transmission and storage. The use of advanced encryption algorithms helps to make data unreadable to attackers, ensuring that only authorized users can access the information [63,64]. **Cryptographic** key management is equally important, as keys are used to encrypt and decrypt data. Proper key management involves securely generating keys, securely distributing them to authorized devices, and periodically rotating keys to mitigate the risks associated with key compro-

mise [65–67]. **Firewalls** are devices or software that monitor and control network traffic by filtering and blocking unauthorized communications. Implementing robust firewalls can help to protect IIoT devices from external intrusions and network attacks [68–70]. **Firmware** and application security is another critical area; as bugs or vulnerabilities in the firmware or software can be exploited by attackers to compromise a device, it is essential that IIoT device manufacturers regularly release firmware and application updates in order to fix known vulnerabilities and protect devices from attacks [71–73]. Finally, security awareness and staff training are important aspects of ensuring that end users are aware of security threats and best practices for protecting their devices and data. Collaboration between vendors, standards organizations, and research institutions is critical to developing and promoting security best practices in the IIoT.

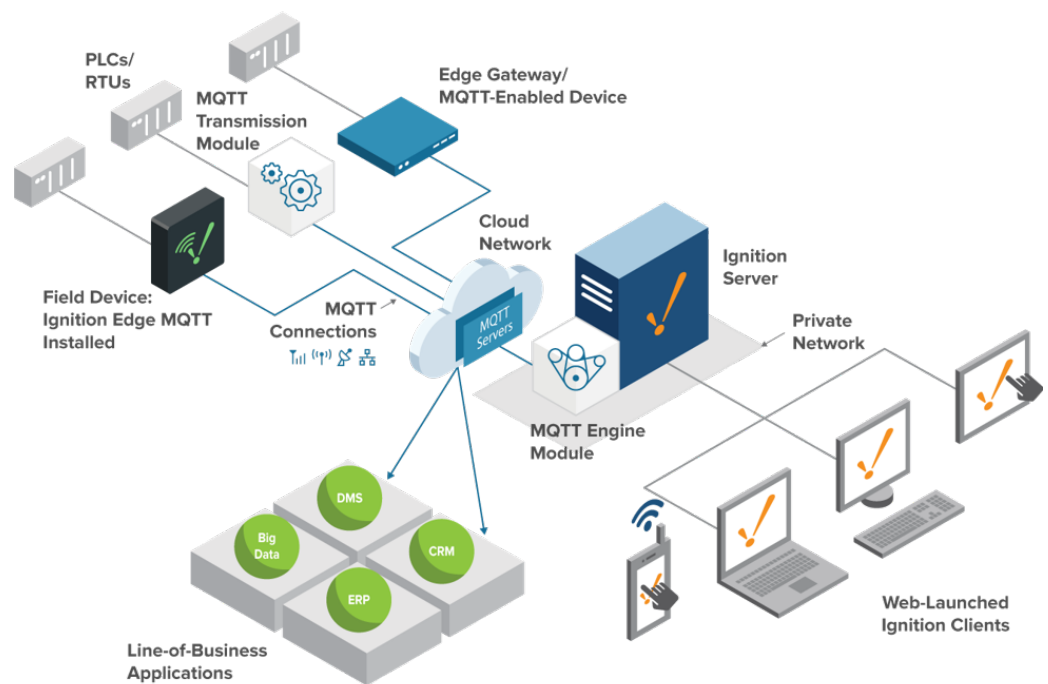


Figure 2. A schematic representation of a typical IIoT communication system.

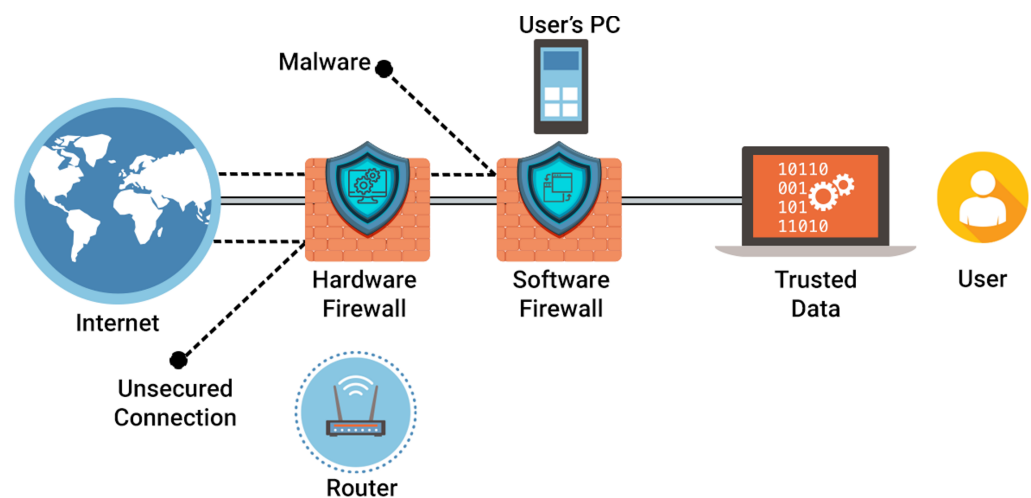


Figure 3. Data safety through HW/SW firewalls.

1.3. Background on AI for the IIoT

Artificial Intelligence (AI) represents a multidisciplinary field of computer science that aims to develop systems and algorithms capable of simulating and replicating human capabilities. The main goal of AI is to develop systems capable of learning from data, drawing conclusions, solving problems, and making decisions autonomously while exploiting a wide range of techniques and methodologies. In the field of artificial intelligence, the recurring keywords which represent the most basic concepts are the following:

- **Machine Learning:** A sub-discipline of AI that focuses on training computers to learn from data without being explicitly programmed. Machine learning algorithms enable computers to identify patterns and relationships in data, allowing them to make predictions or decisions based on new unseen data. This capability makes machine learning particularly useful in tasks such as predictive modeling, classification, clustering, and anomaly detection. By iteratively learning from data, machine learning models can improve their performance over time and adapt to changing conditions [74–77].
- **Artificial Neural Networks:** These are mathematical models inspired by the functioning of the human brain, composed of interconnected artificial neurons organized in layers. Neural networks are capable of learning complex patterns and relationships in data through a process called training. During training, the network adjusts the weights of connections between neurons in order to minimize the difference between the predicted and actual outcomes. Neural networks have demonstrated remarkable performance in various tasks, including image recognition, natural language processing, speech recognition, and time series prediction. Their ability to automatically extract relevant features from raw data makes them a powerful tool in machine learning and AI applications [78–80].
- **Supervised and Unsupervised Learning:** In supervised machine learning, the model is trained on a set of labeled data, where each example is associated with a target variable or outcome. The model learns to map input features to the corresponding target values, enabling it to make predictions on new unseen data. Supervised learning algorithms include regression for predicting continuous outcomes and classification for predicting categorical outcomes. On the other hand, unsupervised machine learning involves training the model on unlabeled data, where the goal is to identify patterns or structures in the data without explicit guidance. Unsupervised learning techniques include clustering, size reduction, and anomaly detection. These methods are valuable for exploring and understanding the underlying structure of data, uncovering hidden patterns, and generating insights without the need for labeled examples [81–83].

In the Industrial Internet of Things (IIoT), AI is used to analyze data generated by connected devices and draw meaningful insights to make intelligent operational decisions. Typical applications include:

- **Fault prediction** is crucial in preventing unplanned and costly downtime in the IIoT space. Using machine learning algorithms, sensor data can be analyzed to identify patterns and warning signals that may indicate imminent failures in industrial equipment. This allows for timely preventive interventions to avoid costly breakdowns, thereby extending the useful life of systems. Traditional preventative maintenance methods can be limited by a lack of real-time data and inability to accurately predict failures. AI overcomes these limitations by enabling more accurate and timely predictive analytics [84–87].
- **Process Optimization** is essential to maximizing efficiency and reducing costs in industrial environments. AI can identify inefficiencies in manufacturing processes by analyzing data in real time and suggesting improvements. This can include production line optimizations, waste reduction, and optimization of processing times. Traditional methods of process optimization can be limited by difficulty in detecting inefficiencies and identifying areas for improvement. AI overcomes these limitations by offering deeper and more proactive analysis of data [88–90].

- **Predictive Maintenance** allows for the prediction of when a plant or machine will require maintenance, thereby avoiding unexpected and costly downtime. By analyzing sensor data in real time, it is possible to detect signs of impending failure and plan preventative interventions before a failure occurs. Traditional methods of scheduled maintenance can be ineffective and expensive, as they rely on fixed maintenance intervals rather than the actual needs of the facility. AI overcomes these limitations by enabling more targeted and data-driven maintenance [91–94].
- **Product Quality Control** is essential in order to ensure that manufacturing standards are met and that products meet customer expectations. AI can monitor product quality by identifying defects or anomalies during the manufacturing process and taking corrective measures in real time. This can improve customer satisfaction, reduce waste, and increase profitability. Traditional quality control methods can be limited by subjectivity and slowness in detecting defects. AI overcomes these limitations by offering objective and immediate analysis of production data [95–97].
- **Cybersecurity** is crucial in the IIoT to protect industrial systems and data from cyber threats and attacks. AI techniques such as anomaly detection and behavior analysis can help to identify suspicious activities and potential security breaches in real time, allowing for timely responses that mitigate risks. Traditional cybersecurity measures may not be sufficient to address the evolving nature of cyber threats in IIoT environments.
AI offers adaptive and proactive cybersecurity solutions to enhance the overall security posture of industrial systems [98–104].
- **Machine Control and Optimization:** With the increasing connectivity of industrial machinery via the IIoT, AI is currently playing a crucial role in enhancing machine control and optimization. By leveraging AI algorithms, real-time data from interconnected machines can be analyzed to optimize machine performance, minimize downtime, and maximize production efficiency. AI-powered control systems can actively adjust machine parameters such as speed, temperature, and pressure to optimize production processes and ensure product quality. Additionally, predictive maintenance algorithms can anticipate machinery failures, allowing for proactive maintenance interventions to prevent costly breakdowns. AI-driven machine control and optimization contribute to overall operational excellence in industrial settings [105–111].

2. Methodology of This Overview

In order to provide a comprehensive and structured survey of deep learning models in Industrial Internet of Things (IIoT) applications, this section outlines the methodology adopted for review. The methodology ensures a systematic approach to selecting, evaluating, and presenting the relevant body of knowledge in the field.

2.1. Scope and Selection Criteria

This survey focuses on the intersection of deep learning models and IIoT applications, specifically targeting models that have shown significant impacts in industrial settings. The key criteria for selecting the models and applications included in this review are:

- **Relevance:** Models and applications that are directly applicable to IIoT scenarios, particularly in areas such as predictive maintenance, quality control, supply chain management, and energy optimization.
- **Impact:** Research and case studies that demonstrate measurable improvements in efficiency, accuracy, or productivity due to the application of deep learning in the IIoT.
- **Novelty:** Inclusion of both well-established models (e.g., CNNs, RNNs, LSTMs) and emerging models (e.g., GANs, autoencoders) to provide a broad perspective on current trends and innovations.
- **Publication Quality:** Preference for peer-reviewed articles, high-impact conference papers, and reputable technical reports to ensure the reliability and validity of the information presented.

2.2. Evaluation Methods

The evaluation of the deep learning models and their applications in the IIoT is conducted using a combination of quantitative and qualitative methods:

- Quantitative Evaluation:
 - Performance Metrics: Analysis of key performance indicators (KPIs) such as accuracy, precision, recall, F1-score, and mean squared error (MSE) to evaluate the effectiveness of deep learning models.
 - Computational Efficiency: Assessment of the computational requirements, including training time, inference speed, and resource consumption, to gauge the practicality of deploying these models in industrial environments.
 - Scalability: Examination of the scalability of models, considering their ability to handle the large-scale data and real-time processing demands typical in IIoT applications.
- Qualitative Evaluation:
 - Applicability: Evaluation of the relevance and applicability of models to various IIoT domains through case studies and real-world implementations.
 - Adaptability: Consideration of the adaptability of models to different industrial contexts and their ability to integrate with existing IIoT infrastructure.
 - Innovative Contributions: Identification of novel contributions and advancements made by the models in improving industrial processes and addressing specific challenges in IIoT.

2.3. Analysis and Synthesis

The collected data were systematically analyzed to identify trends, patterns, and key insights. The analysis involved:

- Comparative analysis of different deep learning models in terms of their performance and applicability in IIoT contexts.
- Synthesis of findings to highlight best practices, challenges, and future directions in the integration of deep learning with IIoT.

By establishing this methodology, we aim to provide a structured and rigorous survey that not only covers the breadth of the field but also offers deep insights into the practical and theoretical aspects of applying deep learning models in IIoT.

3. AI Models in IIoT Applications

In recent years, the integration of deep learning models with the Industrial Internet of Things has revolutionized various industrial sectors. This fusion of advanced technologies optimizes processes, enhances efficiency, and unlocks new productivity. It equips machines to perceive, learn, and make intelligent decisions autonomously, reshaping industrial operations. The IIoT is an interconnected network of physical devices, sensors, and machinery with embedded software, facilitating data exchange and automation in industrial settings. It supports modern industrialization, enabling smart factories and predictive maintenance. Deep learning, a subset of artificial intelligence, allows machines to learn from data, recognize patterns, and make human-like decisions. Embedding deep learning in the IIoT involves sophisticated neural networks in industrial systems that interpret data, detect anomalies, and optimize processes in real time. Applications include predictive maintenance, quality control, supply chain management, and energy optimization. These systems streamline operations and enable predictive and prescriptive analytics, helping businesses to remain competitive. This exploration delves into deep learning models in the context of the IIoT, including an examination of their mechanisms and transformative impact. We aim to demystify these technologies and highlight their potential to revolutionize industrial processes. We invite readers to join us in exploring the intersection of deep learning and the IIoT, where innovation is currently shaping the future of manufacturing and beyond.

3.1. Convolutional Neural Networks

The Convolutional Neural Network (CNN) is a specialized deep neural network designed for visual imagery analysis. It learns spatial hierarchies of features from input data, making it effective for tasks such as computer vision, image and video recognition, and medical image analysis. CNNs are inspired by the visual cortex of animals, where neurons respond to stimuli in specific visual fields. They comprise multiple layers, including convolutional layers, pooling layers, and fully connected layers (see Figure 4). Convolutional layers apply filters to extract features, while pooling layers downsample the feature maps, reducing complexity and enhancing translation invariance. The networks are trained through supervised learning, adjusting parameters to minimize a loss function via techniques like back-propagation. CNNs excel in learning features from raw data, achieving high accuracy in image classification, object detection, and segmentation.

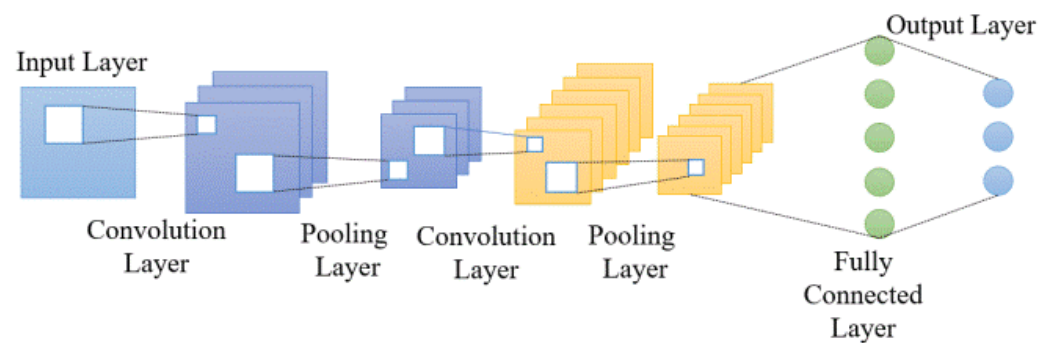


Figure 4. Schematic representation of the typical internal structure of CNNs.

In essence, CNNs process an image by applying filters that create feature maps highlighting various aspects such as edges and textures. Pooling layers then simplify these feature maps, preserving significant features while reducing data volume. Fully connected layers aggregate these features to perform final tasks such as classification or object identification.

CNNs are currently being used to develop effective intrusion detection systems for IIoT networks. A hybrid CNN+LSTM model achieved 93.21% accuracy for binary classification and 92.9% for multi-class classification in detecting attacks on IIoT networks [112,113]. Another approach called IIoT-IDS used an inception CNN model to detect intrusions with high accuracy [114]. Researchers have explored using CNNs for both feature selection and attack detection on IoT networks. The CNN+CNN model, which uses a dual CNN architecture, has been found to outperform other approaches in accuracy, precision, recall, and F1 score for detecting IoT attacks [115]. CNNs combined with edge computing have been applied for IoT-based smart health monitoring systems. Deep CNNs have been proposed along with fog and edge computing to analyze health monitoring tasks, achieving high accuracy in classifying sensor data [116]. CNNs have been used for IoT-based agricultural applications such as pest identification. A crop pest recognition algorithm based on comparing multiple CNN models achieved 99.8% accuracy in distinguishing seven health states of apples and grapes [117]. The ability of CNNs to automatically learn features from raw data makes them well-suited for a variety of IIoT applications that involve processing sensor data, images, and time-series information. As IIoT systems generate massive amounts of data, CNN-based approaches can extract valuable insights, helping to enable intelligent decision making.

3.2. Recurrent Neural Networks

Recurrent Neural Networks (RNNs) are a class of artificial neural networks that are adept at processing sequential data. Unlike feedforward neural networks, RNNs have connections that form directed cycles, enabling them to exhibit dynamic temporal behavior (see Figure 5).

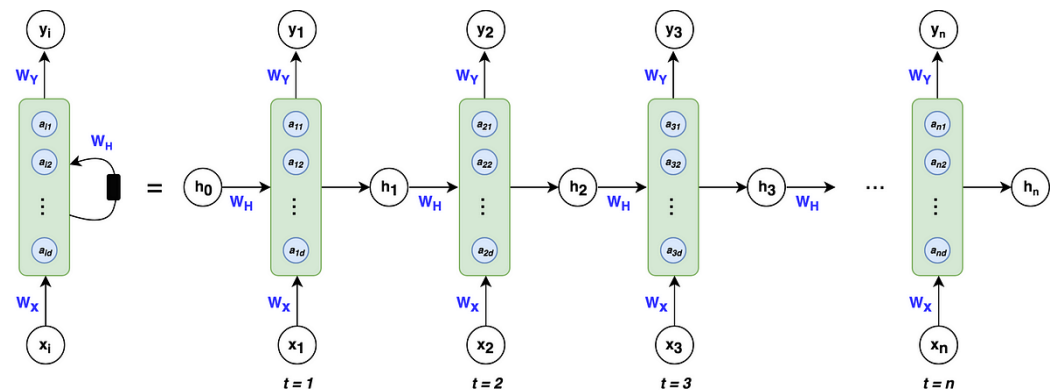


Figure 5. Schematic representation of the typical internal structure of an RNN.

This architecture allows RNNs to capture dependencies and patterns in sequential data such as time series, speech, and text. A key feature of RNNs is their ability to maintain a memory of previous inputs, making them suitable for tasks where context is crucial. However, traditional RNNs suffer from the vanishing gradient problem, which hampers their ability to capture long-term dependencies. To address this, variants such as Long Short-Term Memory (LSTM) networks and Gated Recurrent Units (GRU) have been developed, incorporating mechanisms to better preserve and control the flow of information over time. RNNs find applications in various fields, including natural language processing, speech recognition, time series prediction, and machine translation. They are particularly valuable for anomaly detection systems in IIoT networks, where they have outperformed other approaches in accuracy [118]. LSTM variants have been applied to traffic speed and congestion prediction in large-scale transportation networks, achieving high accuracy [119]. Additionally, RNNs have been combined with other techniques such as machine learning for context-aware intrusion detection in smart factory environments [120]. Despite their potential, implementing RNNs in IIoT faces challenges such as finding an optimal memory size to balance performance and computational complexity.

More research is needed to address these challenges and explore the full potential of RNNs in the IIoT. RNNs, especially LSTM variants, have demonstrated strong performance in IIoT applications involving time series data analysis, anomaly detection, and prediction tasks. As IIoT systems generate massive amounts of sequential data, RNN-based approaches can help to extract valuable insights and enable intelligent decision-making.

3.3. Long Short-Term Memory

Long Short-Term Memory (LSTM) is a type of Recurrent Neural Network (RNN) architecture designed to overcome the limitations of traditional RNNs in capturing long-term dependencies in sequential data.

LSTM networks are equipped with memory cells and various gates, including input, forget, and output gates, which regulate the flow of information through the network. These gates allow LSTM networks to selectively remember or forget information over long sequences, making them particularly effective for tasks such as speech recognition, language modeling, and time series prediction (see Figure 6). LSTM networks process data through a series of steps involving input, forget, and output gates, which manage the information retained or discarded over time. This mechanism enables them to maintain and update a memory of past data points, allowing for the analysis of long-term dependencies. The ability of LSTM networks to retain information over extended periods has made them a popular choice in various fields where analyzing sequential data is crucial for making accurate predictions and decisions, including natural language processing, finance, and healthcare [121]. LSTMs are widely used in IIoT systems for predicting and forecasting time series data. For example, an LSTM-based model was proposed for analyzing industrial IoT equipment and forecasting operation status [122]. Another approach used LSTMs for predicting traffic speeds and congestion in large-scale transportation networks [123].

LSTMs are also applied in anomaly detection within IIoT systems. One study proposed an LSTM–Gauss–NBayes method for outlier detection, building a model on normal time series data and detects outliers by utilizing the predictive error for the Gaussian Naive Bayes model [124]. Additionally, LSTMs are used for condition monitoring and fault detection in industrial systems. For instance, an LSTM-based approach was proposed for fault diagnosis in rotating machines via vibration signals [125]. In smart manufacturing and quality inspection, LSTM models are used to predict non-LOS propagation in wireless sensor networks, which is crucial for accurate location estimation in industrial settings [126]. LSTMs can also be integrated with edge computing for real-time analysis and decision-making in IIoT systems; examples include real-time traffic forecasting in fog computing environments [127].

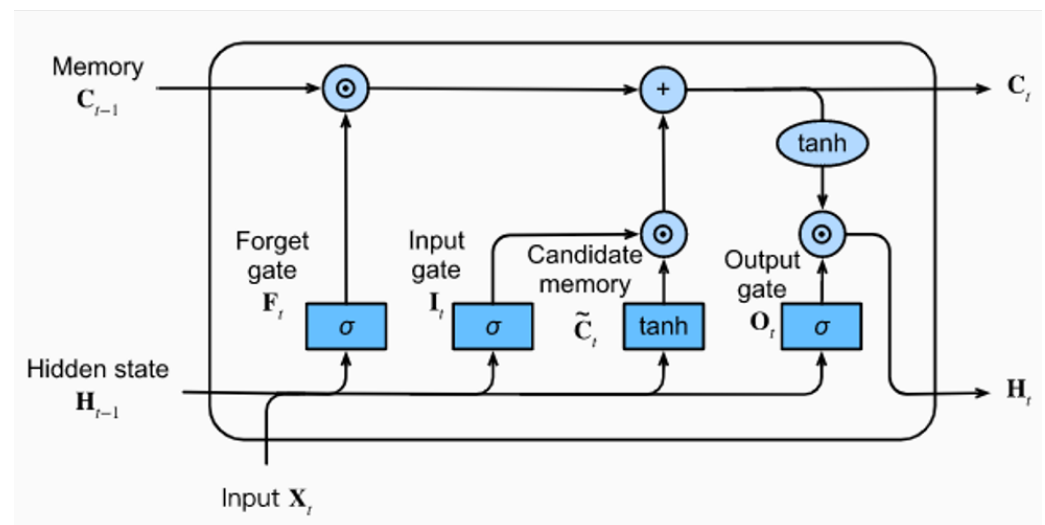


Figure 6. Schematic representation of the internal structure of an LSTM.

3.4. Gated Recurrent Unit

A Gated Recurrent Unit (GRU) is a type of Recurrent Neural Network (RNN) architecture. It addresses the vanishing gradient problem commonly encountered in traditional RNNs by incorporating a gating mechanism that selectively updates and forgets information over time. This mechanism consists of update and reset gates which regulate the flow of information through the network and enable it to capture long-range dependencies in sequential data more effectively. Compared to other RNN variants such as Long Short-Term Memory (LSTM), GRUs offer a simpler architecture with fewer parameters (see Figure 7). Despite this simplicity, they retain strong performance in tasks such as natural language processing, speech recognition, and time series prediction. By striking a balance between capturing relevant information over long sequences and avoiding issues with exploding or vanishing gradients, GRUs have become a popular choice in various deep learning applications [128,129].

In the context of Industrial Internet of Things (IIoT) applications, GRUs are particularly valuable for handling sequential data and learning long-term dependencies. They have been employed in intrusion detection systems for IIoT networks, such as the MAGRU-IDS model, which utilizes a multi-head attention-based GRU to detect anomalies effectively [130]. Furthermore, GRUs contribute to enhanced security in industrial control systems. For example, a sequentially integrated Convolutional-GRU autoencoder has been proposed to bolster security in industrial control systems by leveraging the strengths of both convolutional and recurrent neural networks [131]. Additionally, GRUs are utilized for time series prediction and forecasting in IIoT applications, such as predicting traffic speeds and congestion in large-scale transportation networks [132,133]. They also play a role in condition monitoring and fault detection in industrial systems, including fault diagnosis via vibration signals in rotating machines [134,135]. In the realm of edge computing, GRUs

are integrated for real-time analysis and decision-making in IIoT systems. For instance, an edge computing-based deep learning model utilizing GRUs has been proposed for real-time traffic forecasting [136,137]. Overall, GRUs find wide application in IIoT scenarios involving sequential data, anomaly detection, condition monitoring, and real-time analysis, owing to their ability to learn long-term dependencies and handle sequential data effectively.

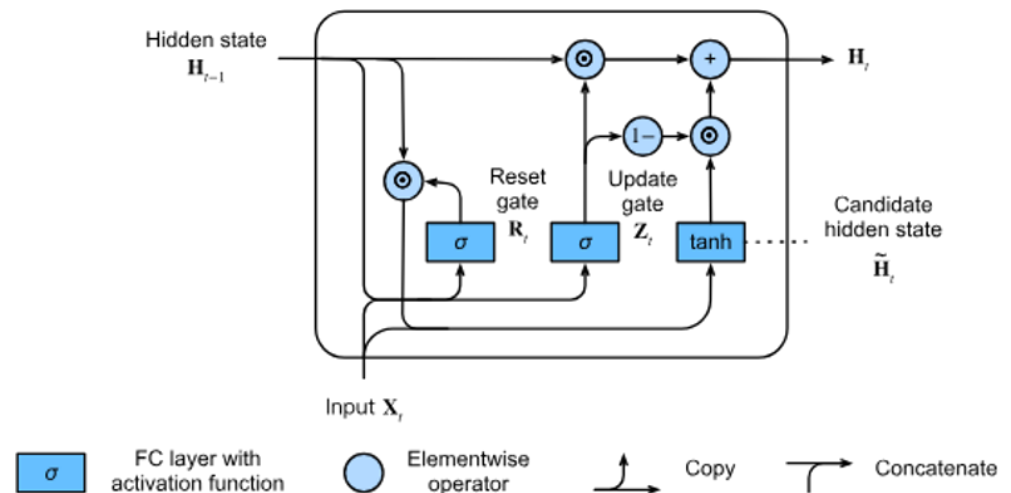


Figure 7. General description of the internal structure of GRU hidden state.

3.5. Generative Adversarial Networks

Generative Adversarial Networks (GANs) are a type of artificial neural network composed of two main components called the a generator and the discriminator. These two components compete in a learning process to create realistic data from scratch (see Figure 8).

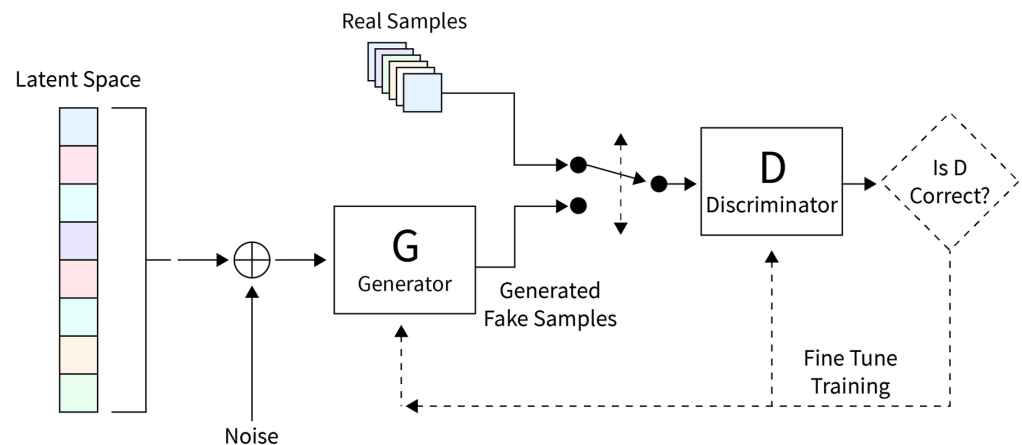


Figure 8. Schematic representation of a GAN model workflow.

The generator is tasked with generating synthetic data, such as images or sounds, from a random set of input data, often referred to as noise vectors. Its goal is to generate data that are indistinguishable from real data. The discriminator, on the other hand, acts as a judge that tries to distinguish between real and generated data. Its task is to provide feedback to the generator on how well it is doing in creating realistic data. Training a GAN occurs iteratively. The generator and discriminator are trained alternately. During each iteration, the generator tries to fool the discriminator by generating increasingly realistic data, while the discriminator seeks to improve in distinguishing between real and generated data. The ultimate goal is for the generator to become so good at creating data that the result

is indistinguishable from real data, while the discriminator becomes increasingly skilled at recognition. When this happens, the GAN reaches a point of equilibrium where the generator has learned to create highly realistic data. GANs find applications in a wide range of fields, such as image processing, speech synthesis, text generation, and much more. They are particularly useful when a large set of real data is not available or when synthetic data need to be generated for testing or training purposes. GANs are also used for intrusion detection and penetration testing in IIoT networks. For instance, a GAN-based approach was proposed for generating synthetic attack samples to train web application layer defensive devices, such as Web Application Firewalls (WAFs) [138], against sophisticated attacks such as Advanced Persistent Threats (APTs) [139,140]. GANs are currently being applied for condition monitoring and fault detection in industrial systems. For example, a GAN-based feedback analysis system was proposed for industrial IoT networks, which can help in detecting anomalies and predicting failures [141]. GANs are being used for time series prediction and forecasting in IIoT applications as well. For instance, a GAN-based model was proposed for predicting traffic speeds and congestion in large-scale transportation networks [142]. GANs can be integrated with edge computing for real-time analysis and decision-making in IIoT systems. For example, a GAN-based approach was proposed for real-time traffic forecasting in fog computing environments [143]. GANs have been applied in various industrial use cases, including semi-supervised learning, image removal, copy detection, and response statistics. They have also been used for anomaly detection and fault diagnosis in industrial systems. A review of GAN applications in Industry 4.0 highlighted the potential of GANs in various industrial use cases, including predictive maintenance, quality control, and supply chain management [144].

3.6. Autoencoder Neural Networks

Autoencoders are a prominent type of deep learning model extensively utilized for unsupervised learning and dimensionality reduction tasks. Structurally, they consist of two primary components: an encoder and a decoder (see Figure 9). The encoder compresses the input data into a lower-dimensional latent-space representation, while the decoder endeavors to reconstruct the original input from this representation.

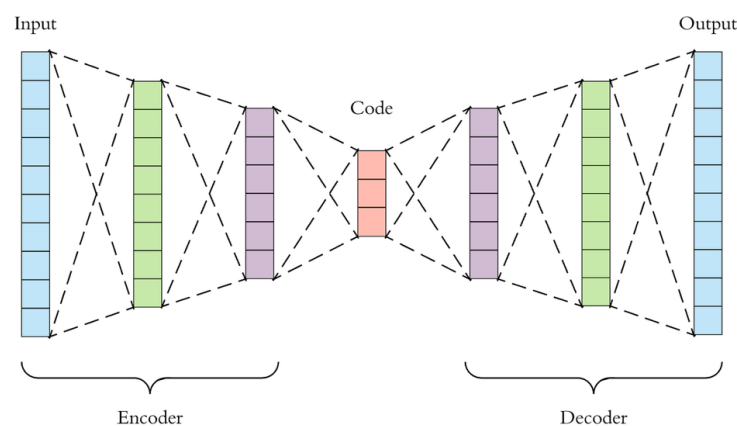


Figure 9. Schematic representation of autoencoder model architecture.

Through this iterative process, autoencoders strive to grasp a compressed and efficient representation of the input data encapsulating its crucial features. They find application across diverse domains, including image and signal processing, anomaly detection, and data denoising. Variants such as denoising autoencoders, sparse autoencoders, and variational autoencoders introduce additional constraints or probabilistic formulations to enhance learning capabilities and generate more meaningful representations. With the surge of deep learning, autoencoders have emerged as fundamental tools for feature learning, data generation, and representation learning tasks. They play a pivotal role in Industrial Internet of Things (IIoT) applications, aiding in tasks such as anomaly detection,

predictive maintenance, and sensor data compression [145]. Autoencoders contribute significantly to anomaly detection in IIoT networks. For instance, a study proposed employing an autoencoder to identify anomalies in industrial IoT data by training the network to disregard noise signals and develop a robust representation of normal data [146]. Furthermore, autoencoders are instrumental in data compression for IIoT applications. For instance, a study employed an autoencoder to compress industrial IoT data, reducing the amount of transmitted and stored data [147]. Moreover, autoencoders play a vital role in dimensionality reduction for IIoT applications. For example, a study utilized an autoencoder to decrease the dimensionality of industrial IoT data, facilitating easier analysis and visualization. Additionally, autoencoders are deployed for condition monitoring and fault detection in industrial systems. For example, a study proposed utilizing an autoencoder to detect faults in industrial equipment by analyzing vibration signals [148,149]. Furthermore, autoencoders are integrated with edge computing for real-time analysis and decision-making in IIoT systems. For example, a study proposed using an autoencoder for real-time anomaly detection in industrial IoT data through edge computing [150,151]. Overall, autoencoders find widespread application in IIoT scenarios encompassing data compression, dimensionality reduction, anomaly detection, and condition monitoring, owing to their ability to learn robust data representations while disregarding noise.

3.7. Summary of AI Model Characteristics

Table 1 provides a comparison of the main deep learning models used in Industrial Internet of Things (IIoT) applications. The table provides an overview of the key features of each model, helping to select the most suitable solution for specific IIoT tasks and data characteristics. Convolutional Neural Networks (CNNs) are particularly effective at analyzing spatially structured data such as images and videos. Convolution allows them to capture local patterns, making them ideal for classification, object detection, and segmentation tasks. However, training CNNs can require a lot of computational resources and abundant training data to avoid overfitting. Recurrent Neural Networks (RNNs), including variants such as LSTM (Long Short-Term Memory) and GRU (Gated Recurrent Unit), are excellent at modeling sequential data such as text, audio, and time series. LSTMs and GRUs address the vanishing gradient problem of traditional RNNs, allowing them to maintain long-term information and handle long sequences. However, RNNs can suffer from stability issues when training on very long sequences, and require careful management of overfitting. Generative Adversarial Networks (GANs) are revolutionary in unsupervised learning, especially for generating realistic data such as images and sounds. The ability of GANs to generate completely new data through competitive training between a generator and a discriminator makes them extremely powerful. However, training GANs can be unstable and requires delicate parameter optimization to avoid problems such as generator collapse and mode collapse. Autoencoders are used for size reduction and data reconstruction, which they accomplish by learning compact representations of input information. They can be useful for image compression, noise filtering, and generating new data instances. However, autoencoders can struggle to capture complex patterns in high-dimensional data, and may not be able to retain useful information during compression.

Table 1. Comparison of deep learning models.

Model	Pros	Cons
CNN	• Effective feature extraction from images. • Efficient handling of large image data volumes.	• Requires large labeled training data. • Not suitable for sequential data.
RNN	• Captures temporal dependencies. • Suitable for time-series forecasting.	• Suffers from vanishing/exploding gradients. • Difficulty capturing long-term dependencies.
LSTM	• Better at capturing long-term dependencies. • Mitigates vanishing/exploding gradients.	• More computationally expensive. • May struggle with very long-term dependencies.
GRU	• Faster training due to fewer parameters. • More computationally efficient.	• May not perform as well as LSTM on some tasks. • Limited ability to capture complex dependencies.
GAN	• Generates realistic synthetic data. • Effective for anomaly detection.	• Unstable training requiring careful tuning. • Limited interpretability of generated data.
Autoenc.	• Learns compact data representations. • Effective for anomaly detection and data reconstruction.	• Performance depends on architecture and hyperparameters. • Limited interpretability of learned representations.

3.8. IIoT Applications for Industry

The integration of Internet of Things (IoT) technology within industrial processes has revolutionized the landscape of manufacturing and production, fostering unprecedented levels of efficiency, automation, and optimization. Through a myriad of applications, the IoT has empowered industries to streamline operations, enhance safety protocols, and optimize resource allocation. One of the most prominent applications lies in predictive maintenance, where IoT sensors embedded in machinery collect real-time data on performance metrics, enabling proactive maintenance schedules based on actual usage patterns rather than predetermined timelines to minimize downtime and reduce costs associated with unexpected breakdowns. Furthermore, the IoT facilitates the implementation of smart energy management systems, allowing industries to monitor energy consumption in real time, identify inefficiencies, and optimize usage to reduce environmental impact and operational costs. Asset tracking and inventory management are also significantly enhanced through IoT solutions, as sensors enable continuous monitoring of inventory levels, location tracking of assets, and real-time updates on stock status, thereby optimizing supply chain logistics and minimizing inventory holding costs. Moreover, IoT-driven quality control mechanisms enable industries to maintain consistent product quality by monitoring and analyzing production parameters in real time, promptly detecting deviations from desired specifications, and making necessary adjustments to ensure adherence to quality standards. In the realm of safety, IoT-enabled systems offer advanced monitoring and alerting capabilities, such as detecting hazardous conditions, monitoring worker vital signs in hazardous environments, and facilitating rapid response to emergencies, thereby mitigating risks and ensuring the well-being of workers. Additionally, the IoT plays a crucial role in enabling the implementation of smart factories, where interconnected devices, machines, and systems communicate and collaborate autonomously, orchestrating production processes with minimal human intervention to achieve higher levels of productivity, agility, and customization. Furthermore, IoT applications extend beyond the confines of the factory floor, encompassing supply chain management, where sensors track the movement and condition of goods throughout the entire supply chain, enabling real-time visibility, traceability, and optimization of logistics processes. In the agricultural sector, IoT solutions empower farmers with actionable insights derived from data collected by sensors deployed across fields, enabling precision agriculture practices such as optimized irrigation, fertilization, and pest management, leading to increased yields, reduced resource consumption, and enhanced sustainability. Additionally, in the transportation and logistics industry, IoT-enabled tracking and monitoring systems enhance fleet management, route optimization, and cargo tracking, improving efficiency, reducing costs, and ensuring timely deliveries. In conclusion, IoT applications in industry encompass a diverse range of functionalities that revolutionize traditional business models, driving efficiencies, improving safety, and

unlocking new opportunities for innovation and growth in the global marketplace. Table 2 summarize the applications of the Industrial IoT.

Table 2. Applications of the Industrial IoT.

Application	Description	Benefits	Challenges	Examples
Predictive Maintenance	Utilizing IoT sensors to monitor the condition of machinery and equipment in real-time, enabling predictive maintenance to prevent costly breakdowns.	- Reduces downtime and maintenance costs - Improves equipment lifespan - Enhances safety by identifying potential failures beforehand	- Requires significant initial investment in IoT sensors and infrastructure - Data security and privacy concerns - Integration with existing systems and workflows	General Electric’s Predix, Siemens MindSphere, Schneider Electric’s EcoStruxure
Asset Tracking and Management	Tracking the location, status, and condition of assets (such as equipment, vehicles, or inventory) using IoT devices and sensors.	- Improved asset utilization and efficiency - Enhanced inventory management and supply chain visibility - Reduction in loss or theft of assets	- Accuracy and reliability of location tracking - Cost of IoT devices and connectivity - Integration with legacy systems and standards	IBM Watson IoT Platform, Cisco Kinetic for Manufacturing, Microsoft Azure IoT Suite
Remote Monitoring and Control	Monitoring and controlling industrial processes, equipment, and systems remotely through IoT-enabled sensors and actuators.	- Enables real-time monitoring and control from anywhere - Reduces the need for on-site personnel - Improves operational efficiency and responsiveness	- Reliability and latency of remote connectivity - Security risks associated with remote access - Compatibility with existing control systems	Honeywell Sentience, ABB Ability, Emerson Plantweb
Quality Control and Assurance	Implementing IoT sensors to monitor and analyze product quality, identify defects, and ensure compliance with quality standards throughout the manufacturing process.	- Improves product quality and consistency - Enables early detection of defects and deviations - Facilitates compliance with regulatory requirements	- Calibration and maintenance of IoT sensors - Integration with quality management systems - Data interpretation and analysis complexities	Bosch IoT Suite, PTC ThingWorx, Rockwell Automation FactoryTalk Analytics
Energy Management and Efficiency	Monitoring and optimizing energy consumption, usage patterns, and efficiency of industrial facilities and equipment through IoT sensors and analytics.	- Reduces energy costs and carbon footprint - Identifies energy-saving opportunities and optimizations - Enhances sustainability initiatives	- Complexities in integrating with existing energy systems - Data accuracy and reliability challenges - Balancing energy efficiency with operational requirements	Sensital iBOTics, General Electric’s Advanced Energy Management System, ABB Energy Management solutions

4. Tools and Devices for Embedded AI in IIoT Applications

In this section, we present several solutions aimed at facilitating the acceleration of Deep Learning (DL) models for embedded applications. The majority of these solutions encompass hardware accelerators meticulously crafted to cater to the exigency of executing DL model inference in low-power environments. Among the showcased solutions, some are tailored for particular application domains, such as the automotive sector (e.g., the Hailo-8

accelerator), while others target broader domains such as space (e.g., the Myriad X, the Google Edge accelerator, and the Nvidia Jetson family). Table 3 provides a comprehensive summary of the solutions delineated in this section.

4.1. Vitis AI

Vitis AI v3.5 is a software solution designed to facilitate AI inference on AMD devices, offering a repository of pretrained AI models and deployment tools spanning edge and data center solutions. Particularly focusing on edge computing scenarios, the utilization of Field-Programmable Gate Arrays (FPGAs) to accelerate Deep Neural Network (DNN) operations emerges as a compelling avenue, capitalizing on the inherent energy efficiency and adaptability afforded by FPGA technology. Figure 10 illustrates the structural layout of the Vitis AI framework.

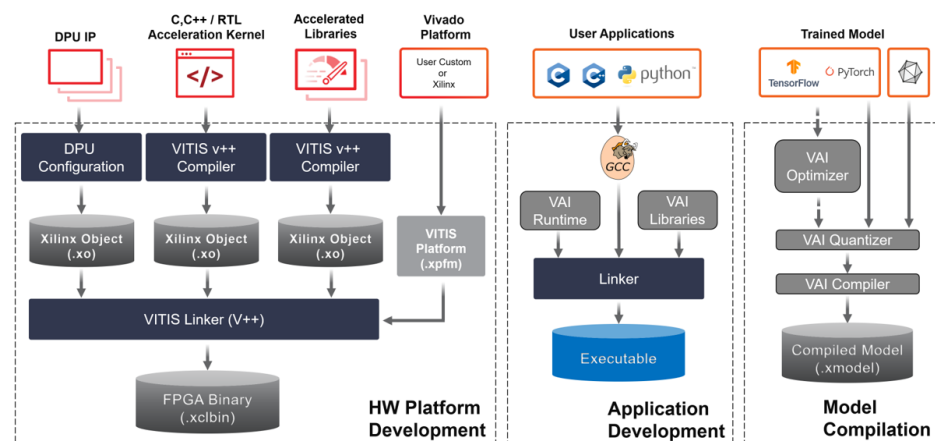


Figure 10. Schematic representation of the Vitis AI framework v3.5 [152].

The Quantizer module serves as a pivotal component, tasked with applying quantization techniques to the model. Utilizing fixed-point representation, such as INT8, for both weights and activations facilitates the reduction of computational complexity and bandwidth usage while enhancing power efficiency. Meanwhile, the Optimizer module endeavors to streamline model complexity through pruning methodologies. Precision pruning, when executed accurately, can effectively curtail computational complexity without incurring significant accuracy losses. The Compiler module assumes responsibility for executing layer fusion and instruction scheduling to maximize parallelism and data reuse, culminating in the generation of instructions for the Deep Learning Processing Unit (DPU). The DPU represents an optimized engine tailored for DNN operations, equipped with a tensor-level instruction set designed to expedite leading-edge Convolutional Neural Networks (CNNs) such as VGG, ResNet, and MobileNet. It extends support to various hardware platforms, including the AMD Zynq UltraScale+™ MPSoCs and the Versal™ AI Edge. Lastly, the Vitis AI Runtime (VART) furnishes a low-level API accessible via both C++ and Python, facilitating seamless integration of the DPU within software applications. Despite numerous endeavors to port neural network models onto FPGA platforms in recent years [153–155], the process remains laborious and often necessitates specialized hardware design expertise. Vitis AI aims to alleviate these challenges by streamlining FPGA-based hardware acceleration, offering an accessible and efficient solution.

4.2. TensorFlow Lite

TensorFlow Lite [156], represents a streamlined iteration of TensorFlow. Employing a flat buffer file format, it accommodates the storage of neural network weights in various formats, including 8-bit integer representations. The conversion process from TensorFlow to TensorFlow Lite capitalizes on quantization and pruning methodologies. Quantization diminishes the bit depth of the model's parameters, such as weights and biases, thereby

reducing the file size of the model. Pruning, on the other hand, endeavors to excise redundant weights, potentially enhancing file compression. The resulting TensorFlow Lite file is executable via the TensorFlow Lite interpreter, which caters to a spectrum of platforms ranging from smartphone operating systems to microcontrollers and other embedded devices.

4.3. TensorFlow Lite for Microcontrollers

Compared to TensorFlow Lite, TensorFlow Lite for Microcontrollers (TensorFlow Lite Micro (TFLM)) [157,158] represents a significant advancement in the realm of low-power devices. Specifically tailored for embedded systems operating on 32-bit platforms such as ARM Cortex-M CPUs, ESP32, and similar architectures, this framework is engineered to facilitate inference with minimal resource overhead. TFLM strives to ensure seamless interoperability across diverse platforms while prioritizing low memory utilization and swift inference times. To deploy projects, the generated code must be integrated into the toolchain corresponding to the targeted device. Leveraging CMSIS-NN [159], TFLM harnesses high-performance inference capabilities on ARM Cortex-M CPUs by exploiting features such as Single Instruction–Multiple Data (SIMD) operations, Digital Signal Processing (DSP) extensions, and ARM’s Machine Learning Vector Extension (MVE). Furthermore, TFLM employs full 8-bit quantization to maximize efficiency and performance in resource-constrained environments. Figure 11 provides a schematic representation of the integration framework between TensorFlow and TensorFlow Lite, illustrating their cohesive operation and compatibility across various platforms. This advanced framework not only extends the capabilities of TensorFlow Lite but also caters to the stringent requirements of embedded systems, facilitating efficient and robust inference in low-power environments.

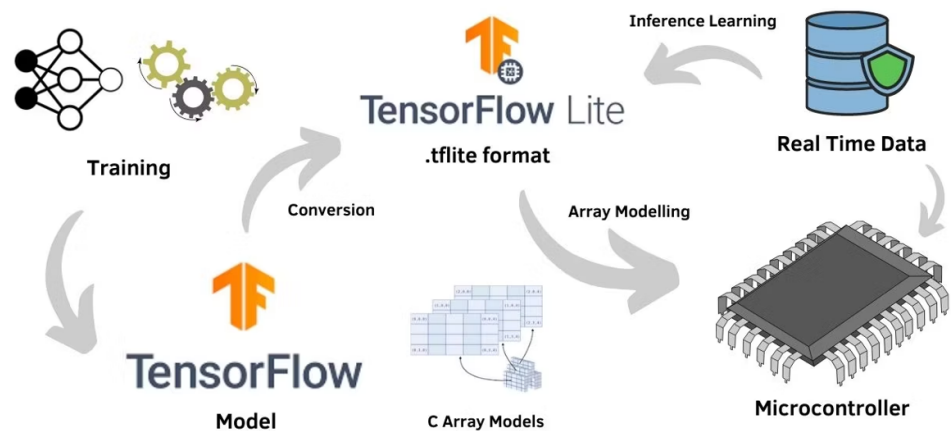


Figure 11. Schematic representation of the Tensorflow /Tensorflow Lite integration framework.

4.4. STM32Cube.AI

STM32Cube.AI, referenced as [160], represents a pivotal tool seamlessly integrated into the STM32CubeMX environment, serving the purpose of deploying DNN models onto STM32 microcontrollers. Users can readily access DNN models from the model zoo [161] provided by ST or import models from third-party frameworks such as TensorFlow and ONNX. Through the ONNX format, STM32Cube.AI extends support to DNN models trained with popular frameworks such as PyTorch v2.3, Matlab v2023b, and Scikit-learn v1.5, ensuring broad compatibility and flexibility. Notably, this tool not only caters to DNN models but also encompasses support for various other machine learning algorithms. Upon model import, STM32Cube.AI generates a C-code library tailored for inclusion in the user’s project, facilitating compilation for STM32 microcontrollers. These microcontrollers typically belong to ARM Cortex-M-based boards, which are characterized by limited flash

and RAM resources and often lack dedicated hardware accelerators for AI algorithms. This underscores the imperative of having an optimized library capable of extracting maximum efficiency and throughput from these devices concerning AI algorithm execution. Furthermore, STM32Cube.AI offers comprehensive functionalities for model validation, providing insights into hardware requirements such as ROM and RAM usage and anticipated inference time, thereby enabling fine-tuning of the trade-off between internal and external memory resource utilization. This feature-rich tool empowers users to optimize AI algorithm performance while adhering to the stringent resource constraints imposed by STM32 microcontrollers. Figure 12 shows the STM32Cube AI framework.

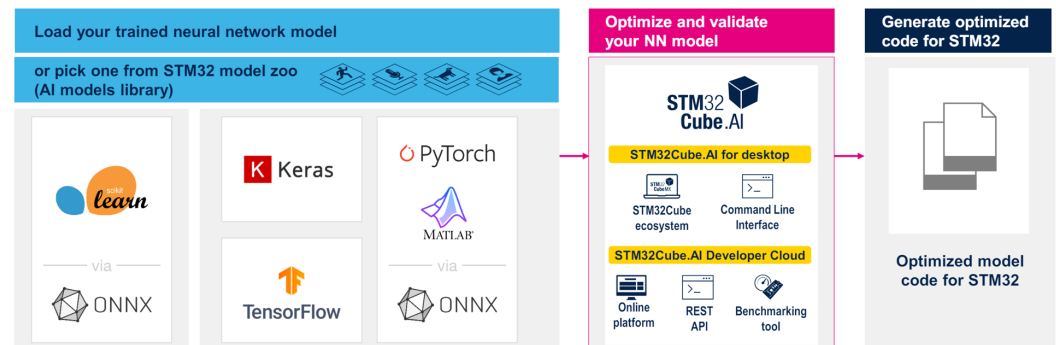


Figure 12. Schematic representation of the STM32Cube AI framework.

4.5. ISPU

STMicroelectronics has developed the Intelligent Sensing Processing Unit (Intelligent Sensor Processing Unit (ISPU)) [162], an ultra-low-power and highly efficient programmable core engineered for executing signal processing and AI algorithms at the edge. Specifically designed to operate within Inertial Measurement Units (Inertial Measurement Unit (IMU)) equipped with MEMS sensors, the ISPU facilitates real-time processing of sensed data with minimal power consumption. Harnessing Digital Signal Processing (Digital Signal Processing (DSP)) capabilities, the ISPU analyzes data using AI algorithms and supports a range of precision levels from full precision to 1-bit Neural Networks (Neural Network (NN)). This energy-efficient solution enables the widespread deployment of AI applications, catering to both industrial and consumer sectors. The ISPU is integrated into expansion boards such as the X-NUCLEO-IKS4A1 board [163], offering versatile programming options through the STM32CubeMX software with the X-CUBE-ISPU expansion. Additionally, NanoEdge AI Studio facilitates the generation of optimized libraries tailored for STM32 devices. This tool supports applications including anomaly detection and multi-class classification, with the resulting library deployable using the X-CUBE-ISPU v2.0 expansion software.

4.6. Renesas E-AI

Renesas has developed e-AI [164], a comprehensive solution tailored to real-time applications running on their range of Microcontroller Units (MCU) and Microprocessor Units (MPU). Renesas offers a variety of devices catering to diverse requirements in terms of AI application performance and resource utilization. To facilitate the deployment of AI models on Renesas devices, the company provides a suite of software tools. These tools enable the conversion of trained models into executables compatible with either the MCU or Renesas' hardware accelerator. For MCU deployment, Renesas offers the e-AI Translator and e-AI Checker. The e-AI Translator converts pretrained models to align with the MCU development environment, while the e-AI Checker computes resource metrics such as RAM, ROM, and execution time based on the selected MCU configuration for NN deployment. Renesas' hardware accelerator, known as DRP-AI, comprises an AI Multiply–Accumulate Processor (AI-MAC) responsible for executing matrix multiplication-based layers along with a Data Re-Processing (DRP) unit to handle generic operations such as pooling and

activation functions. The DRP-AI is designed to efficiently execute both inference and pre/postprocessing tasks while emphasizing low power consumption and flexibility to accommodate future AI model requirements.

4.7. Hailo-8

Hailo has pioneered the Hailo-8 edge AI processor [165], a breakthrough solution designed to facilitate real-time low-latency AI applications on edge devices. Engineered as an expansion module compatible with existing x86- and ARM-based boards running Linux or Windows operating systems, the Hailo-8 processor is equipped with automotive compliance, adhering to ISO-26262 ASIL-B (D) standards [166].

Hailo offers a comprehensive software suite [167] encompassing the following features:

- A dataflow compiler tasked with generating a binary file tailored for the Hailo-8 processor from a pretrained model acquired from third-party high-level frameworks such as TensorFlow v2.15 or PyTorch v2.3. This software suite also incorporates optimizations, including quantization, to maximize the utilization of Hailo-8 hardware resources, alongside profiling information.
- HailoRT, providing C/C++ and Python APIs to enable seamless interaction between host-running applications and the Hailo-8 processor for executing compiled NN models. Additionally, it furnishes a GStreamer element to integrate NN inference seamlessly into a GStreamer pipeline.
- A model zoo [168] housing pretrained models tailored for computer vision tasks.
- A model explorer to assist users in selecting the most suitable models from the model zoo based on specific application requirements such as accuracy and Frames per second (FPS).

The Hailo-8 processor is typically deployed on PCIe boards with M.2 form factor, exemplified by the Hailo-8 M.2 A+E 2230 board. This compact yet powerful solution underscores Hailo's commitment to delivering cutting-edge AI processing capabilities for edge computing applications.

4.8. Google Edge TPU

The Coral Edge Tensor Processing Unit (TPU), developed by Google, is an Application-Specific Integrated Circuit (ASIC) engineered to accelerate TensorFlow Lite models while maintaining exceptionally low power consumption [169]. Specifically optimized for quantized 8-bit NN models compiled for the Edge TPU, it efficiently executes inferences. However, it is noteworthy that the Edge TPU does not support all operations provided by TensorFlow [170]. In such cases, the Edge TPU compiler identifies these unsupported operations and divides the NN into two distinct parts; the initial portion executes on the Edge TPU, while the remainder is offloaded to the CPU. Facilitating streamlined NN inference on the Edge TPU, Google offers the PyCoral API, enabling users to execute complex tasks with minimal lines of Python code [171]. Additionally, Google provides a selection of pretrained NN models tailored for the Edge TPU, encompassing audio classification, object detection, semantic segmentation, and pose estimation tasks [172]. Available in various form factors for both development and production environments, the Google Coral Edge TPU has garnered attention for its suitability in low-Earth-orbit missions [173]. As the space domain witnesses growing demand for low-power Artificial Intelligence (AI) accelerators, the Edge TPU is emerging as a promising solution [169].

4.9. Nvidia Jetson Orin Nano

The Nvidia Jetson Orin Nano [174], an evolution of the original Jetson Nano board, introduces a potent combination of a CPU and an Nvidia Ampere GPU. With configurable power consumption options ranging from 7 to 15 Watts, it is available in a convenient development kit form for rapid prototyping [175]. Equipped with a Linux-based operating system, this versatile platform leverages the computational prowess of its GPU to execute a diverse array of NN models, supporting a wide spectrum of high-level frameworks for

seamless NN inference. Moreover, it harnesses the power of TensorRT [176] to optimize the performance of NN models, ensuring efficient and rapid inference. Notably, certain devices within the Nvidia Jetson family have demonstrated suitability for short-duration satellite missions [177,178], underscoring their resilience and adaptability in challenging environments.

4.10. Intel Movidius Myriad X VPU

The Intel Movidius Myriad X Vision Processing Unit (VPU) is a hardware accelerator renowned for its prowess in executing NN inference using a cluster of processors known as Streaming Hybrid Architecture Vector Engines (SHAVEs). Offering flexibility to developers, it allows customization of the number of SHAVE processors employed for NN inference, with the Myriad X boasting a total of 16 such processors. This versatility empowers users to tailor the inference process, balancing performance and power consumption according to their specific requirements. Demonstrating exceptional performance in accelerating NNs with convolutional layers, such as Fully Convolutional Network (FCN) and Convolutional Neural Network (CNN), the Myriad X stands out as a preferred choice for tasks demanding efficient processing of complex neural networks. Similar to the Google Edge TPU and Nvidia Jetson devices, the Myriad VPU family, encompassing both the Myriad X and its predecessor, the Myriad 2, has found a niche in the space domain. Recognized as a reliable accelerator for NN inference, these devices have been integral to various space missions, including onboard satellites [179] and the International Space Station (ISS) [180,181]. Their adaptability and performance have made them particularly well suited for low-Earth-orbit missions [173,182,183].

4.11. NXP SW and HW Solutions

NXP offers a comprehensive portfolio of microcontrollers (MCUs) and processors tailored for machine learning applications across the automotive, smart industrial, and IoT sectors. Their software development tools, such as the eIQ machine learning environment, enable the utilization of AI algorithms across NXP's extensive range of MCUs and processors [184–186]. NXP has collaborated with NVIDIA to seamlessly integrate NVIDIA TAO APIs within the eIQ machine learning development environment. This integration streamlines the deployment of trained AI models on NXP's edge processing devices. The eIQ machine learning software development environment seamlessly integrates into NXP's MCUXpresso SDK and Yocto Project Linux development environments, empowering developers to create comprehensive system-level applications with ease. NXP's i.MX 93 applications processor boasts a Neural Processing Unit (NPU) capable of executing optimized AI models deployed through the eIQ environment. For MCUs, NXP's i.MX RT Crossover series serves as an ideal platform for embedded AI/ML applications. It supports TensorFlow Lite and Glow neural network models through the eIQ software [187]. The eIQ toolkit provides a comprehensive machine learning workflow tool, enabling the creation, optimization, export, and deployment of ML models and workloads on NXP hardware. Additionally, NXP offers a range of MCUs, processors, and software tools such as eIQ to empower developers in seamlessly integrating and deploying AI models on their embedded hardware platforms, catering to smart applications across the automotive, industrial, and IoT domains. Moving on to NXP's microcontroller families, the S32K1xx and S32K3xx series are engineered to meet the stringent demands of automotive and IoT applications. The S32K1xx family features 32-bit Arm Cortex-M-based MCUs with a basic cryptographic security engine.

They are compliant with ISO 26262 up to ASIL-B for functional safety, and offer scalability in the number of cores, memory, and peripherals for high-performance applications. The S32K3xx family comprises 32-bit Arm Cortex-M7-based MCUs available in single, dual, and lockstep core configurations. These MCUs are compliant with ISO 26262 up to ASIL-D for functional safety, and offer scalability in the number of cores, memory, and peripherals for high-performance applications. Both families are part of NXP's SafeAssure program, ensuring adherence to ISO 26262 standards for system-level safety requirements. They

are also compatible with the NXP S32 Automotive Platform, facilitating seamless software reuse and flexibility across applications in body, zone control, and electrification [188,189].

4.12. Nordic Semiconductor HW Solution

Nordic Semiconductor offers a range of hardware solutions for integrating AI and ML models into their wireless IoT devices. Below, we provide an overview of their available hardware solutions:

- **nRF52 and nRF53 Series Bluetooth SoCs:** These SoCs are now capable of running AI and ML features through a partnership with Edge Impulse, a leading provider of “tiny ML” tools. This integration allows for easy-to-use AI and ML features on resource-constrained wireless IoT chips, making them accessible to a broader range of applications [190].
- **Arm Total Access:** Nordic Semiconductor has adopted Arm Total Access to advance AI and ML capabilities at the edge. This subscription provides advanced access to multiple Arm products, including Cortex CPUs, Ethos NPUs, and the CoreLink System IP. This integration enables Nordic to access greater ML capabilities and computing resources for advanced IoT applications [191].
- **Atlazo Acquisition:** Nordic Semiconductor has acquired the IP portfolio of Atlazo, a US-based technology leader in AI/ML processors, sensor interface design, and energy management for tiny edge devices. This acquisition enhances Nordic’s position in low-power products and solutions for IoT applications and accelerates its strategic development initiatives, particularly in health-related applications [192].

These hardware solutions are designed to support the growing need for AI and ML capabilities at the edge, enabling cross-market applications covering IoT, consumer, and industrial contexts. They provide a comprehensive suite of tools, software, and support for developers to create and deploy AI/ML models on Nordic’s devices.

4.13. Infineon AURIX HW Solutions

Infineon’s AURIX TC2xx and TC3xx microcontrollers cater to automotive and industrial applications, boasting advanced features for connectivity, security, and functional safety [193,194]. The TC2xx family houses up to six independent 32-bit superscalar TriCore V1.6.2 CPUs, while the TC3xx family features a hexa-core high-performance architecture. Both families comply with ISO 26262 up to ASIL-D for functional safety and offer scalability, connectivity, and security features. Suitable for engine management, transmission control, and powertrain applications, they provide robust solutions for demanding automotive environments. The AURIX TC4x family expands Infineon’s offerings, integrating a high-performance AI accelerator called the parallel processing unit (PPU) powered by the Synopsys DesignWare ARC EV Processor IP [195]. This integration enables real-time processing of AI algorithms, including RNNs and CNNs. Supporting functional safety and model-based design, the TC4x family optimizes battery functions, enhances energy efficiency, and ensures safer automotive systems. Collaborations with Synopsys and MathWorks further enrich the development ecosystem, providing optimized tools and software for AI development on AURIX TC4x. In summary, Infineon’s AURIX microcontrollers offer comprehensive hardware solutions for AI integration, ensuring high performance, functional safety, and scalability across automotive and industrial applications.

Table 3. Main solutions for AI on low-power devices and related supported hardware devices.

Name	Supported High Level Frameworks	Supported Hardware	Used Weights Data Types	Typical Applications
Vitis AI	TensorFlow v2.15, PyTorch v2.3, ONNX v1.16.1	Zynq™ UltraScale+™ MPSoC, Versal™ adaptive SoCs, and Alveo™ platforms	INT8	FPGA-based accelerators implementation [196–198]
TensorFlow Lite Micro	TensorFlow Lite v2.15	Microcontrollers-based platforms (e.g., Cortex-M-based platform)	INT8	Real-time compact ML/AI MCU Integration [199–201]
STM32Cube.AI	TensorFlow v2.15, ONNX v1.16.1 (e.g., PyTorch v2.3, Matlab v2023b, and Scikit-learn v1.15)	STM32 microcontrollers	FP32, INT8	Tiny ML/AI for Edge IIoT [202–204]
ISPU	custom C-code algorithms	STM32 microcontrollers	full precision to 1-bit NNs	Integration of In-device ML/AI model for sensors [205,206]
Renesas e-AI	ONNX (e.g., TensorFlow, PyTorch, etc.)	Renesas RZ/V series	INT8	AI-based Device Fingerprinting [207,208]
Hailo-8	Kerasv2.16, TensorFlow v2.15, TensorFlow Lite v2.15, PyTorch v2.3, and ONNX v1.16.1	boards featuring the Hailo processor	INT4-8-16 bits	Accelerated AI-based IIoT systems [209–211]
Edge TPU	TensorFlow Lite compiled for the Edge TPU	boards featuring the Edge TPU	INT8	real-time high-speed computation for edge computing [212,213]
Jetson Orin Nano	Any framework compatible with Nvidia Ampere GPU	Nvidia Jetson Orin Nano w/o development kit	Any data type supported by the Ampere GPU	ML/AI-based Image and Video processing on embedded devices [214–216]
Myriad X	TensorFlow v2.15, Caffe v2.10	Intel Neural Compute Stick 2 and other boards featuring the Myriad X VPU	FP16, fixed point 8-bit	Accelerating AI and Computer Vision for Satellite Applications [180,217]
NXP eIQ	TensorFlow Lite & micro v2.15, Glow v10.9, CMSIS-NN v6.6.0	NXP EdgeVerse MCU and microprocessors (i.e., i.MX RT crossover MCUs, and i.MX family)	FP32, INT8	AI-based Automotive Cybersecurity [218,219]
Nordic Semiconductor	TensorFlow Lite for Microcontrollers v2.15	Nordic Semiconductor nRF5340 and nRF9160 SoCs	INT8	real-time embedded localization systems [220–222]
Aurix	TensorFlow Lite, PyTorch, ONNX	Aurix TC2xx and TC3xx microcontrollers	FP32, INT8	real-time control/monitoring algorithm for automotive [223–225]

5. Conclusions

In this paper, we have emphasized the benefits of using AI for embedded IIoT applications, highlighting several key advantages. One significant benefit is low latency. Unlike server applications that prioritize throughput, embedded applications require extremely low latency in order to be effective. This necessitates AI solutions that can operate within the limited hardware resources of low-power embedded devices. Security and privacy are also crucial when dealing with industrial machinery that must often handle sensitive data. Proper management of these data is essential. Additionally, resilience is vital, as downtime poses a significant risk for industrial applications. The ability to continue operations without an internet connection is a key enabler for AI in IIoT applications. Another important aspect is low power consumption. Embedded applications often target battery-powered devices, making high power efficiency a necessity. We have provided an in-depth analysis of various platforms, including NXP, AURIX, and Nordic Semiconductor. These platforms

offer robust solutions tailored for IIoT applications, combining advanced features with efficient performance. We have summarized the main neural network architectures, software tools, and hardware accelerators that enable AI on embedded low-power devices suitable for IIoT applications. Certain solutions are general-purpose, while others are tailored for specific industrial domains, such as the Myriad VPU for space applications and the Hailo-8 for automotive applications. The techniques and devices introduced in this paper aim to improve various aspects of deploying AI solutions for low-power embedded applications. Software techniques such as quantization reduce model size, saving storage space and allowing faster inference times thanks to a reduced bit width. Many software tools, both open-source and proprietary, facilitate model deployment on embedded devices, thereby reducing time-to-market, which is critical for industrial applications. Hardware solutions focus on maximizing power and cost efficiency while minimizing inference latency. Addressing these challenges is crucial for the development and deployment of AI solutions in embedded IIoT applications. In recent years, the increasing number of AI-based IIoT applications has driven the development of tools and techniques to tackle these challenges. This trend is expected to continue, spurred by the growing adoption of IIoT solutions and the ongoing development of software tools and techniques aimed at simplifying AI deployment on low-power edge devices.

Author Contributions: All authors contributed equally to this research work. All authors have read and agreed to the published version of the manuscript.

Funding: This work has been partially supported by Centro Nazionale di Ricerca in High-Performance Computing Big Data and Quantum Computing SPOKE 6 Multiscale modelling & Engineering applications; and by Project FoReLab MIUR Dipartimenti di Eccellenza.

Data Availability Statement: The data that support the findings of this study are available from the corresponding author (Pierpaolo Dini), upon reasonable request.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Saha, R.; Misra, S.; Deb, P.K. FogFL: Fog-Assisted Federated Learning for Resource-Constrained IoT Devices. *IEEE Internet Things J.* **2021**, *8*, 8456–8463. [\[CrossRef\]](#)
2. Shao, J.; Zhang, J. Communication-Computation Trade-off in Resource-Constrained Edge Inference. *IEEE Commun. Mag.* **2020**, *58*, 20–26. [\[CrossRef\]](#)
3. Mayer, R.; Tariq, M.A.; Rothermel, K. Minimizing communication overhead in window-based parallel complex event processing. In Proceedings of the DEBS '17: The 11th ACM International Conference on Distributed and Event-based Systems, Barcelona, Spain, 19–23 June 2017; pp. 54–65.
4. Pimenov, D.Y.; Mia, M.; Gupta, M.K.; Machado, Á.R.; Pintaude, G.; Unune, D.R.; Khanna, N.; Khan, A.M.; Tomaz, Í.; Wojciechowski, S.; et al. Resource saving by optimization and machining environments for sustainable manufacturing: A review and future prospects. *Renew. Sustain. Energy Rev.* **2022**, *166*, 112660. [\[CrossRef\]](#)
5. Ahmed, Q.W.; Garg, S.; Rai, A.; Ramachandran, M.; Jhanjhi, N.Z.; Masud, M.; Baz, M. Ai-based resource allocation techniques in wireless sensor internet of things networks in energy efficiency with data optimization. *Electronics* **2022**, *11*, 2071. [\[CrossRef\]](#)
6. Yao, C.; Yang, C.; Xiong, Z. Energy-Saving Predictive Resource Planning and Allocation. *IEEE Trans. Commun.* **2016**, *64*, 5078–5095. [\[CrossRef\]](#)
7. Hijji, M.; Ahmad, B.; Alam, G.; Alwakeel, A.; Alwakeel, M.; Abdulaziz Alharbi, L.; Aljarf, A.; Khan, M.U. Cloud servers: Resource optimization using different energy saving techniques. *Sensors* **2022**, *22*, 8384. [\[CrossRef\]](#) [\[PubMed\]](#)
8. Lee, C.; Ahmed, G. Improving IoT privacy, data protection and security concerns. *Int. J. Technol. Innov. Manag. (IJTIM)* **2021**, *1*, 18–33. [\[CrossRef\]](#)
9. Wang, F.; Diao, B.; Sun, T.; Xu, Y. Data Security and Privacy Challenges of Computing Offloading in FINs. *IEEE Netw.* **2020**, *34*, 14–20. [\[CrossRef\]](#)
10. Yang, P.; Xiong, N.; Ren, J. Data Security and Privacy Protection for Cloud Storage: A Survey. *IEEE Access* **2020**, *8*, 131723–131740. [\[CrossRef\]](#)
11. Xu, L.; Jiang, C.; Wang, J.; Yuan, J.; Ren, Y. Information Security in Big Data: Privacy and Data Mining. *IEEE Access* **2014**, *2*, 1149–1176. [\[CrossRef\]](#)
12. Abdelmalak, M.; Benidris, M. Proactive Generation Redispatch to Enhance Power System Resilience During Hurricanes Considering Unavailability of Renewable Energy Sources. *IEEE Trans. Ind. Appl.* **2022**, *58*, 3044–3053. [\[CrossRef\]](#)
13. Jasiūnas, J.; Lund, P.D.; Mikkola, J. Energy system resilience—A review. *Renew. Sustain. Energy Rev.* **2021**, *150*, 111476. [\[CrossRef\]](#)

14. Stanković, A.M.; Tomsovic, K.L.; De Caro, F.; Braun, M.; Chow, J.H.; Čukalevski, N.; Dobson, I.; Eto, J.; Fink, B.; Hachmann, C.; et al. Methods for Analysis and Quantification of Power System Resilience. *IEEE Trans. Power Syst.* **2023**, *38*, 4774–4787. [\[CrossRef\]](#)
15. Ud Din, I.; Bano, A.; Awan, K.A.; Almogren, A.; Altameem, A.; Guizani, M. LightTrust: Lightweight Trust Management for Edge Devices in Industrial Internet of Things. *IEEE Internet Things J.* **2023**, *10*, 2776–2783. [\[CrossRef\]](#)
16. Peniak, P.; Bubeníková, E.; Kanáliková, A. Validation of High-Availability Model for Edge Devices and IIoT. *Sensors* **2023**, *23*, 4871. [\[CrossRef\]](#) [\[PubMed\]](#)
17. Zhang, Y.; Sun, W.; Shi, Y. Architecture and Implementation of Industrial Internet of Things (IIoT) Gateway. In Proceedings of the 2020 IEEE 2nd International Conference on Civil Aviation Safety and Information Technology (ICCASIT), Weihai, China, 14–16 October 2020; pp. 114–120. [\[CrossRef\]](#)
18. Ghosh, A.; Mukherjee, A.; Misra, S. SEGA: Secured Edge Gateway Microservices Architecture for IIoT-Based Machine Monitoring. *IEEE Trans. Ind. Inform.* **2022**, *18*, 1949–1956. [\[CrossRef\]](#)
19. Aazam, M.; Zeadally, S.; Harras, K.A. Deploying Fog Computing in Industrial Internet of Things and Industry 4.0. *IEEE Trans. Ind. Inform.* **2018**, *14*, 4674–4682. [\[CrossRef\]](#)
20. Tange, K.; De Donno, M.; Fafoutis, X.; Dragoni, N. A Systematic Survey of Industrial Internet of Things Security: Requirements and Fog Computing Opportunities. *IEEE Commun. Surv. Tutor.* **2020**, *22*, 2489–2520. [\[CrossRef\]](#)
21. Chalapathi, G.S.S.; Chamola, V.; Vaish, A.; Buyya, R. Industrial internet of things (iiot) applications of edge and fog computing: A review and future directions. In *Fog/Edge Computing for Security, Privacy, and Applications*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 293–325.
22. Vogel, B.; Dong, Y.; Emruli, B.; Davidsson, P.; Spalazzese, R. What is an open IoT platform? Insights from a systematic mapping study. *Future Internet* **2020**, *12*, 73. [\[CrossRef\]](#)
23. Fahmideh, M.; Zowghi, D. An exploration of IoT platform development. *Inf. Syst.* **2020**, *87*, 101409. [\[CrossRef\]](#)
24. Ali, Z.; Mahmood, A.; Khatoon, S.; Alhakami, W.; Ullah, S.S.; Iqbal, J.; Hussain, S. A generic Internet of Things (IoT) middleware for smart city applications. *Sustainability* **2022**, *15*, 743. [\[CrossRef\]](#)
25. Li, J.; Liang, W.; Xu, W.; Xu, Z.; Li, Y.; Jia, X. Service Home Identification of Multiple-Source IoT Applications in Edge Computing. *IEEE Trans. Serv. Comput.* **2023**, *16*, 1417–1430. [\[CrossRef\]](#)
26. Khanna, A.; Kaur, S. Internet of things (IoT), applications and challenges: A comprehensive review. *Wirel. Pers. Commun.* **2020**, *114*, 1687–1762. [\[CrossRef\]](#)
27. Bacco, M.; Boero, L.; Cassara, P.; Colucci, M.; Gotta, A.; Marchese, M.; Patrone, F. IoT Applications and Services in Space Information Networks. *IEEE Wirel. Commun.* **2019**, *26*, 31–37. [\[CrossRef\]](#)
28. Sharma, A.; Babbar, H.; Rani, S.; Sah, D.K.; Sehar, S.; Gianini, G. MHSEER: A meta-heuristic secure and energy-efficient routing protocol for wireless sensor network-based industrial IoT. *Energies* **2023**, *16*, 4198. [\[CrossRef\]](#)
29. Liu, D.; Liang, C.; Mo, H.; Chen, X.; Kong, D.; Chen, P. LEACH-D: A low-energy, low-delay data transmission method for industrial internet of things wireless sensors. *Internet Things-Cyber-Phys. Syst.* **2024**, *4*, 129–137. [\[CrossRef\]](#)
30. Zhang, J.; Yan, Q.; Zhu, X.; Yu, K. Smart industrial IoT empowered crowd sensing for safety monitoring in coal mine. *Digit. Commun. Netw.* **2023**, *9*, 296–305. [\[CrossRef\]](#)
31. Lu, J.; Shen, J.; Vijayakumar, P.; Gupta, B.B. Blockchain-Based Secure Data Storage Protocol for Sensors in the Industrial Internet of Things. *IEEE Trans. Ind. Inform.* **2022**, *18*, 5422–5431. [\[CrossRef\]](#)
32. Liu, Y.; Dillon, T.; Yu, W.; Rahayu, W.; Mostafa, F. Noise Removal in the Presence of Significant Anomalies for Industrial IoT Sensor Data in Manufacturing. *IEEE Internet Things J.* **2020**, *7*, 7084–7096. [\[CrossRef\]](#)
33. Liu, Y.; Dillon, T.; Yu, W.; Rahayu, W.; Mostafa, F. Missing Value Imputation for Industrial IoT Sensor Data With Large Gaps. *IEEE Internet Things J.* **2020**, *7*, 6855–6867. [\[CrossRef\]](#)
34. Gupta, D.; Juneja, S.; Nauman, A.; Hamid, Y.; Ullah, I.; Kim, T.; Tag eldin, E.M.; Ghamry, N.A. Energy Saving Implementation in Hydraulic Press Using Industrial Internet of Things (IIoT). *Electronics* **2022**, *11*, 4061. [\[CrossRef\]](#)
35. Meng, Y.; Li, J. Data sharing mechanism of sensors and actuators of industrial IoT based on blockchain-assisted identity-based cryptography. *Sensors* **2021**, *21*, 6084. [\[CrossRef\]](#) [\[PubMed\]](#)
36. Ma, Y.; Wang, Y.; Cairano, S.D.; Koike-Akino, T.; Guo, J.; Orlik, P.; Guan, X.; Lu, C. Smart Actuation for End-Edge Industrial Control Systems. *IEEE Trans. Autom. Sci. Eng.* **2024**, *21*, 269–283. [\[CrossRef\]](#)
37. Anes, H.; Pinto, T.; Lima, C.; Nogueira, P.; Reis, A. Wearable devices in Industry 4.0: A systematic literature review. In *Proceedings of the International Symposium on Distributed Computing and Artificial Intelligence*; Springer: Berlin/Heidelberg, Germany, 2023; pp. 332–341.
38. Jan, M.A.; Khan, F.; Khan, R.; Mastorakis, S.; Menon, V.G.; Alazab, M.; Watters, P. Lightweight Mutual Authentication and Privacy-Preservation Scheme for Intelligent Wearable Devices in Industrial-CPS. *IEEE Trans. Ind. Inform.* **2021**, *17*, 5829–5839. [\[CrossRef\]](#) [\[PubMed\]](#)
39. Ghafurian, M.; Wang, K.; Dhode, I.; Kapoor, M.; Morita, P.P.; Dautenhahn, K. Smart Home Devices for Supporting Older Adults: A Systematic Review. *IEEE Access* **2023**, *11*, 47137–47158. [\[CrossRef\]](#)
40. Yang, J.; Sun, L. A Comprehensive Survey of Security Issues of Smart Home System: “Spear” and “Shields”, Theory and Practice. *IEEE Access* **2022**, *10*, 124167–124192. [\[CrossRef\]](#)

41. Jmila, H.; Blanc, G.; Shahid, M.R.; Lazrag, M. A Survey of Smart Home IoT Device Classification Using Machine Learning-Based Network Traffic Analysis. *IEEE Access* **2022**, *10*, 97117–97141. [\[CrossRef\]](#)
42. Khan, M.; Silva, B.N.; Han, K. Internet of Things Based Energy Aware Smart Home Control System. *IEEE Access* **2016**, *4*, 7556–7566. [\[CrossRef\]](#)
43. Abbasi, M.; Abbasi, E.; Li, L.; Aguilera, R.P.; Lu, D.; Wang, F. Review on the microgrid concept, structures, components, communication systems, and control methods. *Energies* **2023**, *16*, 484. [\[CrossRef\]](#)
44. Singh, S.V.; Khursheed, A.; Alam, Z. Wired communication technologies and networks for smart grid—A review. In *Cyber Security in Intelligent Computing and Communications*; Springer: Berlin/Heidelberg, Germany, 2022; pp. 183–195.
45. Eid, M.M.; Sorathiya, V.; Lavadiya, S.; Shehata, E.; Rashed, A.N.Z. Free space and wired optics communication systems performance improvement for short-range applications with the signal power optimization. *J. Opt. Commun.* **2021**, 000010151520200304. [\[CrossRef\]](#)
46. Crepaldi, M.; Barcellona, A.; Zini, G.; Ansaldo, A.; Ros, P.M.; Sanginario, A.; Cuccu, C.; Demarchi, D.; Brayda, L. Live Wire—A Low-Complexity Body Channel Communication System for Landmark Identification. *IEEE Trans. Emerg. Top. Comput.* **2021**, *9*, 1248–1264. [\[CrossRef\]](#)
47. Yang, D.; Mahmood, A.; Hassan, S.A.; Gidlund, M. Guest Editorial: Industrial IoT and Sensor Networks in 5G-and-Beyond Wireless Communication. *IEEE Trans. Ind. Inform.* **2022**, *18*, 4118–4121. [\[CrossRef\]](#)
48. Liu, W.; Nair, G.; Li, Y.; Nesic, D.; Vucetic, B.; Poor, H.V. On the Latency, Rate, and Reliability Tradeoff in Wireless Networked Control Systems for IIoT. *IEEE Internet Things J.* **2021**, *8*, 723–733. [\[CrossRef\]](#)
49. Du, R.; Zhen, L. Multiuser physical layer security mechanism in the wireless communication system of the IIOT. *Comput. Secur.* **2022**, *113*, 102559. [\[CrossRef\]](#)
50. Mohsan, S.A.H.; Khan, M.A.; Amjad, H. Hybrid FSO/RF networks: A review of practical constraints, applications and challenges. *Opt. Switch. Netw.* **2023**, *47*, 100697. [\[CrossRef\]](#)
51. Alexandropoulos, G.C.; Shlezinger, N.; Alamzadeh, I.; Imani, M.F.; Zhang, H.; Eldar, Y.C. Hybrid Reconfigurable Intelligent Metasurfaces: Enabling Simultaneous Tunable Reflections and Sensing for 6G Wireless Communications. *IEEE Veh. Technol. Mag.* **2024**, *19*, 75–84. [\[CrossRef\]](#)
52. Chowdhury, M.Z.; Hasan, M.K.; Shahjalal, M.; Hossain, M.T.; Jang, Y.M. Optical Wireless Hybrid Networks: Trends, Opportunities, Challenges, and Research Directions. *IEEE Commun. Surv. Tutor.* **2020**, *22*, 930–966. [\[CrossRef\]](#)
53. Giustina, D.D.; Rinaldi, S. Hybrid Communication Network for the Smart Grid: Validation of a Field Test Experience. *IEEE Trans. Power Deliv.* **2015**, *30*, 2492–2500. [\[CrossRef\]](#)
54. Shi, G.; Shen, X.; Xiao, F.; He, Y. DANTD: A Deep Abnormal Network Traffic Detection Model for Security of Industrial Internet of Things Using High-Order Features. *IEEE Internet Things J.* **2023**, *10*, 21143–21153. [\[CrossRef\]](#)
55. Hewa, T.; Braeken, A.; Liyanage, M.; Ylianttila, M. Fog Computing and Blockchain-Based Security Service Architecture for 5G Industrial IoT-Enabled Cloud Manufacturing. *IEEE Trans. Ind. Inform.* **2022**, *18*, 7174–7185. [\[CrossRef\]](#)
56. Ferrag, M.A.; Friha, O.; Hamouda, D.; Maglaras, L.; Janicke, H. Edge-IIoTset: A New Comprehensive Realistic Cyber Security Dataset of IoT and IIoT Applications for Centralized and Federated Learning. *IEEE Access* **2022**, *10*, 40281–40306. [\[CrossRef\]](#)
57. Wang, J.; Chen, J.; Ren, Y.; Sharma, P.K.; Alfarraj, O.; Tolba, A. Data security storage mechanism based on blockchain industrial Internet of Things. *Comput. Ind. Eng.* **2022**, *164*, 107903. [\[CrossRef\]](#)
58. Xenofontos, C.; Zografopoulos, I.; Konstantinou, C.; Jolfaei, A.; Khan, M.K.; Choo, K.K.R. Consumer, Commercial, and Industrial IoT (In)Security: Attack Taxonomy and Case Studies. *IEEE Internet Things J.* **2022**, *9*, 199–221. [\[CrossRef\]](#)
59. Cai, X.; Geng, S.; Zhang, J.; Wu, D.; Cui, Z.; Zhang, W.; Chen, J. A Sharding Scheme-Based Many-Objective Optimization Algorithm for Enhancing Security in Blockchain-Enabled Industrial Internet of Things. *IEEE Trans. Ind. Inform.* **2021**, *17*, 7650–7658. [\[CrossRef\]](#)
60. Sengupta, J.; Ruj, S.; Bit, S.D. A comprehensive survey on attacks, security issues and blockchain solutions for IoT and IIoT. *J. Netw. Comput. Appl.* **2020**, *149*, 102481. [\[CrossRef\]](#)
61. Yu, X.; Guo, H. A Survey on IIoT Security. In Proceedings of the 2019 IEEE VTS Asia Pacific Wireless Communications Symposium (APWCS), Singapore, 28–30 August 2019; pp. 1–5. [\[CrossRef\]](#)
62. Panchal, A.C.; Khadse, V.M.; Mahalle, P.N. Security Issues in IIoT: A Comprehensive Survey of Attacks on IIoT and Its Countermeasures. In Proceedings of the 2018 IEEE Global Conference on Wireless Computing and Networking (GCWCN), Lonavala, India, 23–24 November 2018; pp. 124–130. [\[CrossRef\]](#)
63. Yu, K.; Tan, L.; Aloqaily, M.; Yang, H.; Jararweh, Y. Blockchain-Enhanced Data Sharing With Traceable and Direct Revocation in IIoT. *IEEE Trans. Ind. Inform.* **2021**, *17*, 7669–7678. [\[CrossRef\]](#)
64. Jia, B.; Zhang, X.; Liu, J.; Zhang, Y.; Huang, K.; Liang, Y. Blockchain-Enabled Federated Learning Data Protection Aggregation Scheme With Differential Privacy and Homomorphic Encryption in IIoT. *IEEE Trans. Ind. Inform.* **2022**, *18*, 4049–4058. [\[CrossRef\]](#)
65. Bader, J.; Michala, A.L. Searchable encryption with access control in industrial internet of things (IIoT). *Wirel. Commun. Mob. Comput.* **2021**, *2021*, 5555362. [\[CrossRef\]](#)
66. Mantravadi, S.; Schnyder, R.; Möller, C.; Brunoe, T.D. Securing IT/OT Links for Low Power IIoT Devices: Design Considerations for Industry 4.0. *IEEE Access* **2020**, *8*, 200305–200321. [\[CrossRef\]](#)
67. Astorga, J.; Barcelo, M.; Urbieto, A.; Jacob, E. Revisiting the feasibility of public key cryptography in light of iiot communications. *Sensors* **2022**, *22*, 2561. [\[CrossRef\]](#)

68. Prasad, S.G.; Sharmila, V.C.; Badrinarayanan, M. Role of Artificial Intelligence based Chat Generative Pre-trained Transformer (ChatGPT) in Cyber Security. In Proceedings of the 2023 2nd International Conference on Applied Artificial Intelligence and Computing (ICAAIC), Salem, India, 4–6 May 2023; pp. 107–114. [\[CrossRef\]](#)
69. Uddin, R.; Kumar, S.A.P. SDN-Based Federated Learning Approach for Satellite-IoT Framework to Enhance Data Security and Privacy in Space Communication. *IEEE J. Radio Freq. Identif.* **2023**, *7*, 424–440. [\[CrossRef\]](#)
70. Ahmadi, S. Next Generation AI-Based Firewalls: A Comparative Study. *Int. J. Comput. (IJC)* **2023**, *49*, 245–262.
71. Sun, P.; Garcia, L.; Salles-Loustau, G.; Zonouz, S. Hybrid Firmware Analysis for Known Mobile and IoT Security Vulnerabilities. In Proceedings of the 2020 50th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), Valencia, Spain, 29 June–2 July 2020; pp. 373–384. [\[CrossRef\]](#)
72. Feng, X.; Zhu, X.; Han, Q.L.; Zhou, W.; Wen, S.; Xiang, Y. Detecting Vulnerability on IoT Device Firmware: A Survey. *IEEE/CAA J. Autom. Sin.* **2023**, *10*, 25–41. [\[CrossRef\]](#)
73. He, D.; Yu, X.; Li, T.; Chan, S.; Guizani, M. Firmware Vulnerabilities Homology Detection Based on Clonal Selection Algorithm for IoT Devices. *IEEE Internet Things J.* **2022**, *9*, 16438–16445. [\[CrossRef\]](#)
74. Dini, P.; Saponara, S. Analysis, design, and comparison of machine-learning techniques for networking intrusion detection. *Designs* **2021**, *5*, 9. [\[CrossRef\]](#)
75. Dini, P.; Colicelli, A.; Saponara, S. Review on Modeling and SOC/SOH Estimation of Batteries for Automotive Applications. *Batteries* **2024**, *10*, 34. [\[CrossRef\]](#)
76. Zhu, X.; Zheng, Q.; Tian, X.; Elhanashi, A.; Saponara, S.; Dini, P. Car Recognition Based on HOG Feature and SVM Classifier. In Proceedings of the International Conference on Applications in Electronics Pervading Industry, Environment and Society; Springer: Berlin/Heidelberg, Germany, 2023; pp. 319–326.
77. Dini, P.; Begni, A.; Ciavarella, S.; De Paoli, E.; Fiorelli, G.; Silvestro, C.; Saponara, S. Design and Testing Novel One-Class Classifier Based on Polynomial Interpolation With Application to Networking Security. *IEEE Access* **2022**, *10*, 67910–67924. [\[CrossRef\]](#)
78. Runge, J.; Zmeureanu, R. Forecasting energy use in buildings using artificial neural networks: A review. *Energies* **2019**, *12*, 3254. [\[CrossRef\]](#)
79. Abdolrasol, M.G.; Hussain, S.S.; Ustun, T.S.; Sarker, M.R.; Hannan, M.A.; Mohamed, R.; Ali, J.A.; Mekhilef, S.; Milad, A. Artificial neural networks based optimization techniques: A review. *Electronics* **2021**, *10*, 2689. [\[CrossRef\]](#)
80. Emambocus, B.A.S.; Jasser, M.B.; Amphawan, A. A Survey on the Optimization of Artificial Neural Networks Using Swarm Intelligence Algorithms. *IEEE Access* **2023**, *11*, 1280–1294. [\[CrossRef\]](#)
81. Jafari, F.; Dorafshan, S. Comparison between Supervised and Unsupervised Learning for Autonomous Delamination Detection Using Impact Echo. *Remote Sens.* **2022**, *14*, 6307. [\[CrossRef\]](#)
82. Chen, Y.; Mancini, M.; Zhu, X.; Akata, Z. Semi-Supervised and Unsupervised Deep Visual Learning: A Survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2024**, *46*, 1327–1347. [\[CrossRef\]](#) [\[PubMed\]](#)
83. Gwilliam, M.; Shrivastava, A. Beyond supervised vs. unsupervised: Representative benchmarking and analysis of image representation learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 21–24 June 2022; pp. 9642–9652.
84. Huang, H.; Ding, S.; Zhao, L.; Huang, H.; Chen, L.; Gao, H.; Ahmed, S.H. Real-Time Fault Detection for IIoT Facilities Using GBRBM-Based DNN. *IEEE Internet Things J.* **2020**, *7*, 5713–5722. [\[CrossRef\]](#)
85. Jarwar, M.A.; Khowaja, S.A.; Dev, K.; Adhikari, M.; Hakak, S. NEAT: A Resilient Deep Representational Learning for Fault Detection Using Acoustic Signals in IIoT Environment. *IEEE Internet Things J.* **2023**, *10*, 2864–2871. [\[CrossRef\]](#)
86. Lang, W.; Hu, Y.; Gong, C.; Zhang, X.; Xu, H.; Deng, J. Artificial Intelligence-Based Technique for Fault Detection and Diagnosis of EV Motors: A Review. *IEEE Trans. Transp. Electrification* **2022**, *8*, 384–406. [\[CrossRef\]](#)
87. Elhanashi, A.; Saponara, S.; Zheng, Q. Classification and Localization of Multi-Type Abnormalities on Chest X-Rays Images. *IEEE Access* **2023**, *11*, 83264–83277. [\[CrossRef\]](#)
88. da Silva, A.; Gil, M.M. Industrial processes optimization in digital marketplace context: A case study in ornamental stone sector. *Results Eng.* **2020**, *7*, 100152. [\[CrossRef\]](#)
89. Jiang, J.; Zu, Y.; Li, X.; Meng, Q.; Long, X. Recent progress towards industrial rhamnolipids fermentation: Process optimization and foam control. *Bioresour. Technol.* **2020**, *298*, 122394. [\[CrossRef\]](#) [\[PubMed\]](#)
90. Liu, W.; Huang, G.; Zheng, A.; Liu, J. Research on the optimization of IIoT data processing latency. *Comput. Commun.* **2020**, *151*, 290–298. [\[CrossRef\]](#)
91. Begni, A.; Dini, P.; Saponara, S. Design and test of an LSTM-based algorithm for Li-Ion batteries remaining useful life estimation. In Proceedings of the International Conference on Applications in Electronics Pervading Industry, Environment and Society; Springer: Berlin/Heidelberg, Germany, 2022; pp. 373–379.
92. Dini, P.; Basso, G.; Saponara, S.; Romano, C. Real-time monitoring and ageing detection algorithm design with application on SiC-based automotive power drive system. *IET Power Electron.* **2024**. [\[CrossRef\]](#)
93. Dini, P.; Ariaudo, G.; Botto, G.; Greca, F.L.; Saponara, S. Real-time electro-thermal modelling and predictive control design of resonant power converter in full electric vehicle applications. *IET Power Electron.* **2023**, *16*, 2045–2064. [\[CrossRef\]](#)
94. Pacini, F.; Dini, P.; Fanucci, L. Design of an Assisted Driving System for Obstacle Avoidance Based on Reinforcement Learning Applied to Electrified Wheelchairs. *Electronics* **2024**, *13*, 1507. [\[CrossRef\]](#)

95. Anderson, H.E.; Santos, I.C.; Hildenbrand, Z.L.; Schug, K.A. A review of the analytical methods used for beer ingredient and finished product analysis and quality control. *Anal. Chim. Acta* **2019**, *1085*, 1–20. [[CrossRef](#)] [[PubMed](#)]
96. Pang, J.; Zhang, N.; Xiao, Q.; Qi, F.; Xue, X. A new intelligent and data-driven product quality control system of industrial valve manufacturing process in CPS. *Comput. Commun.* **2021**, *175*, 25–34. [[CrossRef](#)]
97. Wang, T.; Chen, Y.; Qiao, M.; Snoussi, H. A fast and robust convolutional neural network-based defect detection model in product quality control. *Int. J. Adv. Manuf. Technol.* **2018**, *94*, 3465–3471. [[CrossRef](#)]
98. Rosadini, C.; Chiarelli, S.; Nesci, W.; Saponara, S.; Gagliardi, A.; Dini, P. Method for Protection from Cyber Attacks to a Vehicle Based upon Time Analysis, and Corresponding Device. U.S. Patent 17/929,370, 16 March 2023.
99. Rosadini, C.; Chiarelli, S.; Cornelio, A.; Nesci, W.; Saponara, S.; Dini, P.; Gagliardi, A. Method for Protection from Cyber Attacks to a Vehicle Based Upon Time Analysis, and Corresponding Device. U.S. Patent 18/163,488, 3 August 2023.
100. Dini, P.; Saponara, S. Design and Experimental Assessment of Real-Time Anomaly Detection Techniques for Automotive Cybersecurity. *Sensors* **2023**, *23*, 9231. [[CrossRef](#)] [[PubMed](#)]
101. Elhanashi, A.; Dini, P.; Saponara, S.; Zheng, Q. Integration of Deep Learning into the IoT: A Survey of Techniques and Challenges for Real-World Applications. *Electronics* **2023**, *12*, 4925. [[CrossRef](#)]
102. Elhanashi, A.; Saponara, S.; Dini, P.; Zheng, Q.; Morita, D.; Raytchev, B. An integrated and real-time social distancing, mask detection, and facial temperature video measurement system for pandemic monitoring. *J. -Real-Time Image Process.* **2023**, *20*, 95. [[CrossRef](#)]
103. Elhanashi, A.; Gasmi, K.; Begni, A.; Dini, P.; Zheng, Q.; Saponara, S. Machine learning techniques for anomaly-based detection system on CSE-CIC-IDS2018 dataset. In *Proceedings of the International Conference on Applications in Electronics Pervading Industry, Environment and Society*; Springer: Berlin/Heidelberg, Germany, 2022; pp. 131–140.
104. Dini, P.; Elhanashi, A.; Begni, A.; Saponara, S.; Zheng, Q.; Gasmi, K. Overview on Intrusion Detection Systems Design Exploiting Machine Learning for Networking Cybersecurity. *Appl. Sci.* **2023**, *13*, 7507. [[CrossRef](#)]
105. Pacini, F.; Di Matteo, S.; Dini, P.; Fanucci, L.; Bucchi, F. Innovative Plug-and-Play System for Electrification of Wheel-Chairs. *IEEE Access* **2023**, *11*, 89038–89051. [[CrossRef](#)]
106. Dini, P.; Saponara, S. Model-based design of an improved electric drive controller for high-precision applications based on feedback linearization technique. *Electronics* **2021**, *10*, 2954. [[CrossRef](#)]
107. Dini, P.; Saponara, S. Design of an observer-based architecture and non-linear control algorithm for cogging torque reduction in synchronous motors. *Energies* **2020**, *13*, 2077. [[CrossRef](#)]
108. Dini, P.; Saponara, S. Design of adaptive controller exploiting learning concepts applied to a BLDC-based drive system. *Energies* **2020**, *13*, 2512. [[CrossRef](#)]
109. Dini, P.; Saponara, S. Processor-in-the-loop validation of a gradient descent-based model predictive control for assisted driving and obstacles avoidance applications. *IEEE Access* **2022**, *10*, 67958–67975. [[CrossRef](#)]
110. Bernardeschi, C.; Dini, P.; Domenici, A.; Mouhagir, A.; Palmieri, M.; Saponara, S.; Sassolas, T.; Zaourar, L. Co-simulation of a model predictive control system for automotive applications. In *Proceedings of the International Conference on Software Engineering and Formal Methods*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 204–220.
111. Benedetti, D.; Agnelli, J.; Gagliardi, A.; Dini, P.; Saponara, S. Design of a digital dashboard on low-cost embedded platform in a fully electric vehicle. In *Proceedings of the 2020 IEEE International Conference on Environment and Electrical Engineering and 2020 IEEE Industrial and Commercial Power Systems Europe (EEEIC/I&CPS Europe)*, Madrid, Spain, 9–12 June 2020; pp. 1–5.
112. Xu, X.; Sun, J.; Wang, C.; Zou, B. A novel hybrid CNN-LSTM compensation model against DoS attacks in power system state estimation. *Neural Process. Lett.* **2022**, *54*, 1597–1621. [[CrossRef](#)]
113. Abdallah, M.; An Le Khac, N.; Jahromi, H.; Delia Jurcut, A. A hybrid CNN-LSTM based approach for anomaly detection systems in SDNs. ARES 2021: The 16th International Conference on Availability, Reliability and Security, Vienna, Austria, 17–20 August 2021; pp. 1–7.
114. Alkahtani, H.; Aldhyani, T.H. Botnet attack detection by using CNN-LSTM model for Internet of Things applications. *Secur. Commun. Netw.* **2021**, *2021*, 3806459. [[CrossRef](#)]
115. Alabsi, B.A.; Anbar, M.; Rihan, S.D.A. CNN-CNN: Dual Convolutional Neural Network Approach for Feature Selection and Attack Detection on Internet of Things Networks. *Sensors* **2023**, *23*, 6507. [[CrossRef](#)] [[PubMed](#)]
116. Alonazi, M.; Ansar, H.; Mudawi, N.A.; Alotaibi, S.S.; Almujally, N.A.; Alazeb, A.; Jalal, A.; Kim, J.; Min, M. Smart Healthcare Hand Gesture Recognition Using CNN-Based Detector and Deep Belief Network. *IEEE Access* **2023**, *11*, 84922–84933. [[CrossRef](#)]
117. Latif, G.; Abdelhamid, S.E.; Mallouhy, R.E.; Alghazo, J.; Kazimi, Z.A. Deep learning utilization in agriculture: Detection of rice plant diseases using an improved CNN model. *Plants* **2022**, *11*, 2230. [[CrossRef](#)]
118. Ullah, I.; Mahmoud, Q.H. Design and Development of RNN Anomaly Detection Model for IoT Networks. *IEEE Access* **2022**, *10*, 62722–62750. [[CrossRef](#)]
119. Kim, Y.; Wang, P.; Mihaylova, L. Structural Recurrent Neural Network for Traffic Speed Prediction. In *Proceedings of the ICASSP 2019—2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Barcelona, Spain, 12–17 May 2019; pp. 5207–5211. [[CrossRef](#)]
120. Pech, M.; Vrchota, J.; Bednář, J. Predictive maintenance and intelligent sensors in smart factory. *Sensors* **2021**, *21*, 1470. [[CrossRef](#)] [[PubMed](#)]

121. Wang, X.; Yang, L.T.; Cao, E.; Guo, L.; Ren, L.; Deen, M.J. A Tensor-based t-SVD-LSTM Remaining Useful Life Prediction Model for Industrial Intelligence. *IEEE Trans. Ind. Inform.* **2022**, 1–12. [\[CrossRef\]](#)
122. Zhang, W.; Guo, W.; Liu, X.; Liu, Y.; Zhou, J.; Li, B.; Lu, Q.; Yang, S. LSTM-Based Analysis of Industrial IoT Equipment. *IEEE Access* **2018**, 6, 23551–23560. [\[CrossRef\]](#)
123. Ranjan, N.; Bhandari, S.; Zhao, H.P.; Kim, H.; Khan, P. City-Wide Traffic Congestion Prediction Based on CNN, LSTM and Transpose CNN. *IEEE Access* **2020**, 8, 81606–81620. [\[CrossRef\]](#)
124. Wu, D.; Jiang, Z.; Xie, X.; Wei, X.; Yu, W.; Li, R. LSTM Learning With Bayesian and Gaussian Processing for Anomaly Detection in Industrial IoT. *IEEE Trans. Ind. Inform.* **2020**, 16, 5244–5253. [\[CrossRef\]](#)
125. Wang, T.; Zhang, L.; Wang, X. Fault Detection for Motor Drive Control System of Industrial Robots Using CNN-LSTM-based Observers. *CES Trans. Electr. Mach. Syst.* **2023**, 7, 144–152. [\[CrossRef\]](#)
126. Kim, D.H.; Farhad, A.; Pyun, J.Y. UWB Positioning System Based on LSTM Classification With Mitigated NLOS Effects. *IEEE Internet Things J.* **2023**, 10, 1822–1835. [\[CrossRef\]](#)
127. Hu, L.; Miao, Y.; Yang, J.; Ghoneim, A.; Hossain, M.S.; Alrashoud, M. IF-RANs: Intelligent Traffic Prediction and Cognitive Caching toward Fog-Computing-Based Radio Access Networks. *IEEE Wirel. Commun.* **2020**, 27, 29–35. [\[CrossRef\]](#)
128. Wang, Y.; Liao, W.; Chang, Y. Gated recurrent unit network-based short-term photovoltaic forecasting. *Energies* **2018**, 11, 2163. [\[CrossRef\]](#)
129. Brandão Lent, D.M.; Novaes, M.P.; Carvalho, L.F.; Lloret, J.; Rodrigues, J.J.P.C.; Proença, M.L. A Gated Recurrent Unit Deep Learning Model to Detect and Mitigate Distributed Denial of Service and Portscan Attacks. *IEEE Access* **2022**, 10, 73229–73242. [\[CrossRef\]](#)
130. Ullah, S.; Boulila, W.; Koubâa, A.; Ahmad, J. MAGRU-IDS: A Multi-Head Attention-Based Gated Recurrent Unit for Intrusion Detection in IIoT Networks. *IEEE Access* **2023**, 11, 114590–114601. [\[CrossRef\]](#)
131. Hussain, B.Z.; Khan, I. Sequentially Integrated Convolutional-Gated Recurrent Unit Autoencoder for Enhanced Security in Industrial Control Systems. *TechRxiv* **2024**. [\[CrossRef\]](#) [\[PubMed\]](#)
132. Bellocchi, L.; Geroliminis, N. Unraveling reaction-diffusion-like dynamics in urban congestion propagation: Insights from a large-scale road network. *Sci. Rep.* **2020**, 10, 4876. [\[CrossRef\]](#) [\[PubMed\]](#)
133. Ma, X.; Dai, Z.; He, Z.; Ma, J.; Wang, Y.; Wang, Y. Learning traffic as images: A deep convolutional neural network for large-scale transportation network speed prediction. *Sensors* **2017**, 17, 818. [\[CrossRef\]](#) [\[PubMed\]](#)
134. Hong, K.; Pan, J.; Jin, M. Transformer Condition Monitoring Based on Load-Variied Vibration Response and GRU Neural Networks. *IEEE Access* **2020**, 8, 178685–178694. [\[CrossRef\]](#)
135. Su, X.; Shan, Y.; Li, C.; Mi, Y.; Fu, Y.; Dong, Z. Spatial-temporal attention and GRU based interpretable condition monitoring of offshore wind turbine gearboxes. *IET Renew. Power Gener.* **2022**, 16, 402–415. [\[CrossRef\]](#)
136. Liu, X.; Zhang, W.; Zhou, X.; Zhou, Q. MECGuard: GRU enhanced attack detection in Mobile Edge Computing environment. *Comput. Commun.* **2021**, 172, 1–9. [\[CrossRef\]](#)
137. Huang, X.; Yuan, Y.; Chang, C.; Gao, Y.; Zheng, C.; Yan, L. Human Activity Recognition Method Based on Edge Computing-Assisted and GRU Deep Learning Network. *Appl. Sci.* **2023**, 13, 9059. [\[CrossRef\]](#)
138. Chowdhary, A.; Jha, K.; Zhao, M. Generative Adversarial Network (GAN)-Based Autonomous Penetration Testing for Web Applications. *Sensors* **2023**, 23, 8014. [\[CrossRef\]](#)
139. Gan, C.; Lin, J.; Huang, D.W.; Zhu, Q.; Tian, L. Advanced persistent threats and their defense methods in industrial Internet of things: A survey. *Mathematics* **2023**, 11, 3115. [\[CrossRef\]](#)
140. Li, Y.; Dai, W.; Bai, J.; Gan, X.; Wang, J.; Wang, X. An Intelligence-Driven Security-Aware Defense Mechanism for Advanced Persistent Threats. *IEEE Trans. Inf. Forensics Secur.* **2019**, 14, 646–661. [\[CrossRef\]](#)
141. Yu, W.; Sun, Y.; Zhou, R.; Liu, X. GAN Based Method for Labeled Image Augmentation in Autonomous Driving. In Proceedings of the 2019 IEEE International Conference on Connected Vehicles and Expo (ICCVE), Graz, Austria, 4–8 November 2019; pp. 1–5. [\[CrossRef\]](#)
142. Lee, J.; Shiotsuka, D.; Nishimori, T.; Nakao, K.; Kamijo, S. Gan-based lidar translation between sunny and adverse weather for autonomous driving and driving simulation. *Sensors* **2022**, 22, 5287. [\[CrossRef\]](#) [\[PubMed\]](#)
143. Zhang, M.; Zhang, Y.; Zhang, L.; Liu, C.; Khurshid, S. Deeproad: Gan-based metamorphic testing and input validation framework for autonomous driving systems. In Proceedings of the 33rd ACM/IEEE International Conference on Automated Software Engineering, Montpellier, France, 3–7 September 2018; pp. 132–142.
144. Ma, C.T.; Gu, Z.H. Review of GaN HEMT applications in power converters over 500 W. *Electronics* **2019**, 8, 1401. [\[CrossRef\]](#)
145. Tien, C.W.; Huang, T.Y.; Chen, P.C.; Wang, J.H. Using autoencoders for anomaly detection and transfer learning in IoT. *Computers* **2021**, 10, 88. [\[CrossRef\]](#)
146. Torabi, H.; Mirtaheri, S.L.; Greco, S. Practical autoencoder based anomaly detection by using vector reconstruction error. *Cybersecurity* **2023**, 6, 1. [\[CrossRef\]](#)
147. Liu, T.; Wang, J.; Liu, Q.; Alibhai, S.; Lu, T.; He, X. High-ratio lossy compression: Exploring the autoencoder to compress scientific data. *IEEE Trans. Big Data* **2021**, 9, 22–36. [\[CrossRef\]](#)
148. Yang, L.; Zhang, Z. A Conditional Convolutional Autoencoder-Based Method for Monitoring Wind Turbine Blade Breakages. *IEEE Trans. Ind. Inform.* **2021**, 17, 6390–6398. [\[CrossRef\]](#)

149. Roy, M.; Bose, S.K.; Kar, B.; Gopalakrishnan, P.K.; Basu, A. A Stacked Autoencoder Neural Network based Automated Feature Extraction Method for Anomaly detection in On-line Condition Monitoring. In Proceedings of the 2018 IEEE Symposium Series on Computational Intelligence (SSCI), Bangalore, India, 18–21 November 2018; pp. 1501–1507. [CrossRef]
150. Lee, S.J.; Yoo, P.D.; Asyhari, A.T.; Jhi, Y.; Chermak, L.; Yeun, C.Y.; Taha, K. IMPACT: Impersonation Attack Detection via Edge Computing Using Deep Autoencoder and Feature Abstraction. *IEEE Access* **2020**, *8*, 65520–65529. [CrossRef]
151. Yu, W.; Liu, Y.; Dillon, T.; Rahayu, W. Edge Computing-Assisted IoT Framework With an Autoencoder for Fault Detection in Manufacturing Predictive Maintenance. *IEEE Trans. Ind. Inform.* **2023**, *19*, 5701–5710. [CrossRef]
152. Xilinx. Vitis AI. Available online: <https://www.xilinx.com/products/design-tools/vitis/vitis-ai.html> (accessed on 1 June 2024).
153. Nannipieri, P.; Giuffrida, G.; Diana, L.; Panicacci, S.; Zulberti, L.; Fanucci, L.; Hernandez, H.G.M.; Hubner, M. Icu4sat: A general-purpose reconfigurable instrument control unit based on open source components. In Proceedings of the 2022 IEEE Aerospace Conference (AERO), Big Sky, MT, USA, 5–12 March 2022; pp. 1–9.
154. Pacini, T.; Rapuano, E.; Fanucci, L. Fpg-ai: A technology-independent framework for the automation of cnn deployment on fpgas. *IEEE Access* **2023**, *11*, 32759–32775. [CrossRef]
155. Mittal, S. A survey of FPGA-based accelerators for convolutional neural networks. *Neural Comput. Appl.* **2020**, *32*, 1109–1139. [CrossRef]
156. Google. TensorFlow Lite. Available online: <https://www.tensorflow.org/lite> (accessed on 1 June 2024).
157. David, R.; Duke, J.; Jain, A.; Reddi, V.J.; Jeffries, N.; Li, J.; Kreeger, N.; Nappier, I.; Natraj, M.; Regev, S.; et al. Tensorflow lite micro: Embedded machine learning on tinymml systems. *arXiv* **2020**, arXiv:2010.08678.
158. Google. TensorFlow Lite for Microcontrollers. Available online: <https://www.tensorflow.org/lite/microcontrollers> (accessed on 1 June 2024).
159. CMSIS. CMSIS-NN. Available online: https://arm-software.github.io/CMSIS_6/latest/NN/index.html (accessed on 1 June 2024).
160. STMicroelectronics. STM32Cube.AI. Available online: <https://stm32ai.st.com/stm32-cube-ai/> (accessed on 1 June 2024).
161. STMicroelectronics. AI Model Zoo for STM32 Devices. Available online: <https://github.com/STMicroelectronics/stm32ai-modelzoo/> (accessed on 1 June 2024).
162. STMicroelectronics. IMUs with Intelligent Sensor Processing Unit. Available online: https://www.st.com/content/st_com/en/campaigns/ispu-ai-in-sensors.html (accessed on 1 June 2024).
163. STMicroelectronics. X-NUCLEO-IKS4A1 Expansion Board for STM32 Nucleo. Available online: <https://www.st.com/en/ecosystems/x-nucleo-iks4a1.html> (accessed on 1 June 2024).
164. Renesas. E-AI Solutions. Available online: <https://www.renesas.com/us/en/key-technologies/artificial-intelligence/e-ai> (accessed on 1 June 2024).
165. Hailo. Hailo-8 for Edge Devices. Available online: <https://hailo.ai/products/ai-accelerators/hailo-8-ai-accelerator/#hailo8-overview> (accessed on 1 June 2024).
166. Bahig, G.; El-Kadi, A. Formal verification of automotive design in compliance with ISO 26262 design verification guidelines. *IEEE Access* **2017**, *5*, 4505–4516. [CrossRef]
167. Hailo. Hailo Software Suite for AI Applications. Available online: <https://hailo.ai/products/hailo-software/hailo-ai-software-suite/#sw-overview> (accessed on 1 June 2024).
168. Hailo. Hailo Model Zoo. Available online: https://github.com/hailo-ai/hailo_model_zoo/tree/master (accessed on 1 June 2024).
169. Google. Edge TPU. Available online: <https://coral.ai/products/> (accessed on 1 June 2024).
170. Google. TensorFlow Models on the Edge TPU. Available online: <https://coral.ai/docs/edgetpu/models-intro> (accessed on 1 June 2024).
171. Google. Run Inference on the Edge TPU with Python. Available online: <https://coral.ai/docs/edgetpu/tflite-python/> (accessed on 1 June 2024).
172. Google. Models for Edge TPU. Available online: <https://coral.ai/models/> (accessed on 1 June 2024).
173. Ramaswami, D.P.; Hiemstra, D.M.; Yang, Z.W.; Shi, S.; Chen, L. Single event upset characterization of the intel movidius myriad x vpu and google edge tpu accelerators using proton irradiation. In Proceedings of the 2022 IEEE Radiation Effects Data Workshop (REDW) (in Conjunction with 2022 NSREC), Provo, UT, USA, 18–22 July 2022; pp. 1–3.
174. Nvidia. Jetson Orin for Next-Gen Robotics. Available online: <https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/jetson-orin/> (accessed on 1 June 2024).
175. Nvidia. Jetson Orin Nano Developer Kit Getting Started. Available online: <https://developer.nvidia.com/embedded/learn/get-started-jetson-orin-nano-devkit> (accessed on 1 June 2024).
176. Nvidia. TensorRT SDK. Available online: <https://developer.nvidia.com/tensorrt> (accessed on 1 June 2024).
177. Slater, W.S.; Tiwari, N.P.; Lovelley, T.M.; Mee, J.K. Total ionizing dose radiation testing of NVIDIA Jetson nano GPUs. In Proceedings of the 2020 IEEE High Performance Extreme Computing Conference (HPEC), Boston, MA, USA, 21–25 September 2020; pp. 1–3.
178. Rad, I.O.; Alarcia, R.M.G.; Dengler, S.; Golkar, A.; Manfletti, C. Preliminary Evaluation of Commercial Off-The-Shelf GPUs for Machine Learning Applications in Space. Semester Thesis, Technical University of Munich, Munich, Germany, 6 September 2023.

179. Giuffrida, G.; Diana, L.; de Gioia, F.; Benelli, G.; Meoni, G.; Donati, M.; Fanucci, L. CloudScout: A deep neural network for on-board cloud detection on hyperspectral images. *Remote Sens.* **2020**, *12*, 2205. [CrossRef]
180. Dunkel, E.; Swope, J.; Towfic, Z.; Chien, S.; Russell, D.; Sauvageau, J.; Sheldon, D.; Romero-Cañas, J.; Espinosa-Aranda, J.L.; Buckley, L.; et al. Benchmarking deep learning inference of remote sensing imagery on the qualcomm snapdragon and intel movidius myriad x processors onboard the international space station. In Proceedings of the IGARSS 2022-2022 IEEE International Geoscience and Remote Sensing Symposium, Kuala Lumpur, Malaysia, 17–22 July 2022; pp. 5301–5304.
181. Dunkel, E.R.; Swope, J.; Candela, A.; West, L.; Chien, S.A.; Towfic, Z.; Buckley, L.; Romero-Cañas, J.; Espinosa-Aranda, J.L.; Hervas-Martin, E.; et al. Benchmarking Deep Learning Models on Myriad and Snapdragon Processors for Space Applications. *J. Aerosp. Inf. Syst.* **2023**, *20*, 660–674. [CrossRef]
182. Furano, G.; Meoni, G.; Dunne, A.; Moloney, D.; Ferlet-Cavrois, V.; Tavoularis, A.; Byrne, J.; Buckley, L.; Psarakis, M.; Voss, K.O.; et al. Towards the Use of Artificial Intelligence on the Edge in Space Systems: Challenges and Opportunities. *IEEE Aerosp. Electron. Syst. Mag.* **2020**, *35*, 44–56. [CrossRef]
183. Buckley, L.; Dunne, A.; Furano, G.; Tali, M. Radiation test and in orbit performance of mp soc ai accelerator. In Proceedings of the 2022 IEEE Aerospace Conference (AERO), Big Sky, MT, USA, 5–12 March 2022; pp. 1–9.
184. Chappa, R.T.N.; El-Sharkawy, M. Deployment of SE-SqueezeNext on NXP BlueBox 2.0 and NXP i.MX RT1060 MCU. In Proceedings of the 2020 IEEE Midwest Industry Conference (MIC), Champaign, IL, USA, 7–8 August 2020; Volume 1, pp. 1–4. [CrossRef]
185. Desai, S.R.; Sinha, D.; El-Sharkawy, M. Image Classification on NXP i.MX RT1060 using Ultra-thin MobileNet DNN. In Proceedings of the 2020 10th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, 6–8 January 2020; pp. 474–480. [CrossRef]
186. Ayi, M.; El-Sharkawy, M. Real-time Implementation of RMNv2 Classifier in NXP Bluebox 2.0 and NXP i.MX RT1060. In Proceedings of the 2020 IEEE Midwest Industry Conference (MIC), Champaign, IL, USA, 7–8 August 2020; Volume 1, pp. 1–4. [CrossRef]
187. NXP-eIQ ML Software Development Environment. Available online: <https://www.nxp.com/design/design-center/software/eiq-ml-development-environment:EIQ> (accessed on 12 June 2024).
188. NXP. S32K1 Microcontrollers for Automotive General Purpose. 2020. Available online: <https://www.nxp.com/products/processors-and-microcontrollers/s32-automotive-platform/s32k-auto-general-purpose-mcus/s32k1-microcontrollers-for-automotive-general-purpose:S32K1> (accessed on 5 June 2024).
189. NXP. S33K1 Microcontrollers for Automotive General Purpose. 2022. Available online: <https://www.nxp.com/products/processors-and-microcontrollers/s32-automotive-platform/s32k-auto-general-purpose-mcus/s32k3-microcontrollers-for-automotive-general-purpose:S32K3> (accessed on 5 June 2024).
190. Semiconductor, N. Bluetooth Low Energy and Bluetooth Mesh Development Kit for the nRF52810 and nRF52832 SoCs. 2022. Available online: <https://www.nordicsemi.com/Products/Development-hardware/nRF52-DK> (accessed on 5 June 2024).
191. ARM. Arm Total Access—Accelerate Development and Time-to-Market. 2022. Available online: <https://www.arm.com/products/licensing/arm-total-access> (accessed on 5 June 2024).
192. Semiconductor, N. Nordic to Acquire AI/ML Technology in the US. 2022. Available online: <https://www.nordicsemi.com/Nordic-news/2023/08/Nordic-to-acquire-AI-ML-technology-in-the-US> (accessed on 5 June 2024).
193. Infineon, A. 32-bit TriCore™ AURIX™—TC2xx. 2022. Available online: <https://www.infineon.com/cms/en/product/microcontroller/32-bit-tricore-microcontroller/32-bit-tricore-aurix-tc2xx/> (accessed on 5 June 2024).
194. Infineon, A. 32-bit TriCore™ AURIX™—TC3xx. 2022. Available online: https://www.infineon.com/cms/en/product/microcontroller/32-bit-tricore-microcontroller/32-bit-tricore-aurix-tc3xx/?_gl=1*16hlusy*_up*MQ..&gclid=CjwKCAjwmYCzBhA6EiwAxFwfgBRAswH3Tly-AZ6-ADyxjsCXa2yu8Dey2HkYe-zmKJidyweXxgvghoCm4wQAvD_BwE&gclsrc=aw.ds (accessed on 5 June 2024).
195. Infineon, A. 32-bit TriCore™ AURIX™—TC4xx. 2022. Available online: https://www.infineon.com/cms/en/product/microcontroller/32-bit-tricore-microcontroller/32-bit-tricore-aurix-tc4x/?_gl=1*yej4gb*_up*MQ..&gclid=CjwKCAjwmYCzBhA6EiwAxFwfgBRAswH3Tly-AZ6-ADyxjsCXa2yu8Dey2HkYe-zmKJidyweXxgvghoCm4wQAvD_BwE&gclsrc=aw.ds (accessed on 5 June 2024).
196. Wang, J.; Gu, S. FPGA Implementation of Object Detection Accelerator Based on Vitis-AI. In Proceedings of the 2021 11th International Conference on Information Science and Technology (ICIST), Chengdu, China, 21–23 May 2021; pp. 571–577. [CrossRef]
197. Kathail, V. Xilinx vitis unified software platform. In Proceedings of the FPGA '20: The 2020 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays, Seaside, CA, USA, 23–25 February 2020; pp. 173–174.
198. Ushiroyama, A.; Watanabe, M.; Watanabe, N.; Nagoya, A. Convolutional neural network implementations using Vitis AI. In Proceedings of the 2022 IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, 26–29 January 2022; pp. 0365–0371. [CrossRef]
199. Sallang, N.C.A.; Islam, M.T.; Islam, M.S.; Arshad, H. A CNN-Based Smart Waste Management System Using TensorFlow Lite and LoRa-GPS Shield in Internet of Things Environment. *IEEE Access* **2021**, *9*, 153560–153574. [CrossRef]

200. Labrèche, G.; Evans, D.; Marszk, D.; Mladenov, T.; Shiradhonkar, V.; Soto, T.; Zelenevskiy, V. OPS-SAT Spacecraft Autonomy with TensorFlow Lite, Unsupervised Learning, and Online Machine Learning. In Proceedings of the 2022 IEEE Aerospace Conference (AERO), Big Sky, MT, USA, 5–12 March 2022; pp. 1–17. [\[CrossRef\]](#)
201. Manor, E.; Greenberg, S. Custom Hardware Inference Accelerator for TensorFlow Lite for Microcontrollers. *IEEE Access* **2022**, *10*, 73484–73493. [\[CrossRef\]](#)
202. De Vita, F.; Nocera, G.; Bruneo, D.; Tomaselli, V.; Falchetto, M. On-Device Training of Deep Learning Models on Edge Microcontrollers. In Proceedings of the 2022 IEEE International Conferences on Internet of Things (iThings) and IEEE Green Computing & Communications (GreenCom) and IEEE Cyber, Physical & Social Computing (CPSCom) and IEEE Smart Data (SmartData) and IEEE Congress on Cybermatics (Cybermatics), Espoo, Finland, 22–25 August 2022; pp. 62–69. [\[CrossRef\]](#)
203. Akhtari, S.; Pickhardt, F.; Pau, D.; Pietro, A.D.; Tomarchio, G. Intelligent Embedded Load Detection at the Edge on Industry 4.0 Powertrains Applications. In Proceedings of the 2019 IEEE 5th International forum on Research and Technology for Society and Industry (RTSI), Florence, Italy, 9–12 September 2019; pp. 427–430. [\[CrossRef\]](#)
204. Crocioni, G.; Pau, D.; Delorme, J.M.; Gruosso, G. Li-Ion Batteries Parameter Estimation With Tiny Neural Networks Embedded on Intelligent IoT Microcontrollers. *IEEE Access* **2020**, *8*, 122135–122146. [\[CrossRef\]](#)
205. Pau, D.P.; Randriatsimiovalaza, M.D. Electromyography Gestures Sensing with Deeply Quantized Neural Networks. In Proceedings of the 2023 IEEE International Conference on Metrology for eXtended Reality, Artificial Intelligence and Neural Engineering (MetroXRaine), Milano, Italy, 25–27 October 2023; pp. 711–716. [\[CrossRef\]](#)
206. Ronco, A.; Schulthess, L.; Zehnder, D.; Magno, M. Machine Learning In-Sensors: Computation-enabled Intelligent Sensors For Next Generation of IoT. In Proceedings of the 2022 IEEE Sensors, Dallas, TX, USA, 30 October–2 November 2022; pp. 1–4. [\[CrossRef\]](#)
207. Hung, C.W.; Wu, J.R.; Lee, C.H. Device Light Fingerprints Identification Using MCU-Based Deep Learning Approach. *IEEE Access* **2021**, *9*, 168134–168140. [\[CrossRef\]](#)
208. Safi, M.; Dadkhah, S.; Shoeleh, F.; Mahdikhani, H.; Molyneaux, H.; Ghorbani, A.A. A survey on IoT profiling, fingerprinting, and identification. *ACM Trans. Internet Things* **2022**, *3*, 1–39. [\[CrossRef\]](#)
209. Kim, R.; Kim, J.; Yoo, H.; Kim, S.C. Implementation of deep learning based intelligent image analysis on an edge AI platform using heterogeneous AI accelerators. In Proceedings of the 2023 14th International Conference on Information and Communication Technology Convergence (ICTC), Jeju Island, Korea, 11–13 October 2023; pp. 1347–1349.
210. Mika, K.; Griessl, R.; Kucza, N.; Porrmann, F.; Kaiser, M.; Tigges, L.; Hagemeyer, J.; Trancoso, P.; Azhar, M.W.; Qararyah, F.; et al. VEDLIoT: Next generation accelerated AIoT systems and applications. In Proceedings of the CF '23: 20th ACM International Conference on Computing Frontiers, Bologna, Italy, 9–11 May 2023; pp. 291–296.
211. Griessl, R.; Porrmann, F.; Kucza, N.; Mika, K.; Hagemeyer, J.; Kaiser, M.; Porrmann, M.; Tassemeier, M.; Flottmann, M.; Qararyah, F.; et al. Evaluation of heterogeneous AIoT Accelerators within VEDLIoT. In Proceedings of the 2023 Design, Automation & Test in Europe Conference & Exhibition (DATE), Ingrid Verbauwhede, KU Leuven, Leuven, Belgium, 13 November 2023; pp. 1–6. [\[CrossRef\]](#)
212. Sengupta, J.; Kubendran, R.; Neftci, E.; Andreou, A. High-Speed, Real-Time, Spike-Based Object Tracking and Path Prediction on Google Edge TPU. In Proceedings of the 2020 2nd IEEE International Conference on Artificial Intelligence Circuits and Systems (AICAS), Genova, Italy, 31 August–2 September 2020; pp. 134–135. [\[CrossRef\]](#)
213. Seshadri, K.; Akin, B.; Laudon, J.; Narayanaswami, R.; Yazdanbakhsh, A. An Evaluation of Edge TPU Accelerators for Convolutional Neural Networks. In Proceedings of the 2022 IEEE International Symposium on Workload Characterization (IISWC), Austin, TX, USA, 6–8 November 2022; pp. 79–91. [\[CrossRef\]](#)
214. Barnell, M.; Raymond, C.; Smiley, S.; Isereau, D.; Brown, D. Ultra Low-Power Deep Learning Applications at the Edge with Jetson Orin AGX Hardware. In Proceedings of the 2022 IEEE High Performance Extreme Computing Conference (HPEC), Virtually, 19–23 September 2022; pp. 1–4. [\[CrossRef\]](#)
215. Pham, H.V.; Tran, T.G.; Le, C.D.; Le, A.D.; Vo, H.B. Benchmarking Jetson Edge Devices with an End-to-end Video-based Anomaly Detection System. In *Future of Information and Communication Conference*; Springer: Berlin/Heidelberg, Germany, 2024; pp. 358–374.
216. Alexey, G.; Klyachin, V.; Eldar, K.; Driaba, A. Autonomous mobile robot with AI based on Jetson Nano. In *Future Technologies Conference (FTC) 2020*; Springer: Berlin/Heidelberg, Germany, 2021; Volume 1, pp. 190–204.
217. Leon, V.; Minaidis, P.; Lentaris, G.; Soudris, D. Accelerating AI and Computer Vision for Satellite Pose Estimation on the Intel Myriad X Embedded SoC. *Microprocess. Microsyst.* **2023**, *103*, 104947. [\[CrossRef\]](#)
218. Bajer, M. Securing and Hardening Embedded Linux Devices—Case study based on NXP i.MX6 Platform. In Proceedings of the 2022 9th International Conference on Future Internet of Things and Cloud (FiCloud), Rome, Italy (and Online), 22–24 August 2022; pp. 181–189. [\[CrossRef\]](#)
219. Pathak, D.; El-Sharkawy, M. Architecturally Compressed CNN: An Embedded Realtime Classifier (NXP Bluebox2.0 with RTMaps). In Proceedings of the 2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, 7–9 January 2019; pp. 331–336. [\[CrossRef\]](#)
220. Cao, Y.F.; Cheung, S.W.; Yuk, T.I. A Multiband Slot Antenna for GPS/WiMAX/WLAN Systems. *IEEE Trans. Antennas Propag.* **2015**, *63*, 952–958. [\[CrossRef\]](#)
221. Bajaj, R.; Ranaweera, S.; Agrawal, D. GPS: Location-tracking technology. *Computer* **2002**, *35*, 92–94. [\[CrossRef\]](#)

222. Takai, M.; Martin, J.; Bagrodia, R.; Ren, A. Directional virtual carrier sensing for directional antennas in mobile ad hoc networks. In Proceedings of the MobiHoc02: ACM Symposium on Mobile Ad Hoc Networking and Networking, Lausanne, Switzerland, 9–11 June 2002; pp. 183–193.
223. Díaz, E.; Mezzetti, E.; Kosmidis, L.; Abella, J.; Cazorla, F.J. Modelling multicore contention on the aurix tm tc27x. In Proceedings of the DAC '18: The 55th Annual Design Automation Conference 2018, San Francisco, CA, USA, 24–29 June 2018; pp. 1–6.
224. Mezzetti, E.; Barbina, L.; Abella, J.; Botta, S.; Cazorla, F.J. AURIX TC277 Multicore Contention Model Integration for Automotive Applications. In Proceedings of the 2019 Design, Automation & Test in Europe Conference & Exhibition (DATE), Florence, Italy, 25–29 March 2019; pp. 1202–1203. [[CrossRef](#)]
225. Azad, F.; Islam, Y.; Md Ruslan, C.Z.; Aye Mong Marma, C.; Kalpoma, K.A. Efficient Lane Detection and Keeping for Autonomous Vehicles in Real-World Scenarios. In Proceedings of the 2023 26th International Conference on Computer and Information Technology (ICCIT), Cox's Bazar, Bangladesh, 13–15 December 2023; pp. 1–6. [[CrossRef](#)]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.