future internet

Review

Edge and Cloud Computing in Smart Cities

Maria Trigka * and Elias Dritsas

Department of Informatics and Computer Engineering, University of West Attica, 12243 Egaleo, Greece; dritsase@ceid.upatras.gr
* Correspondence: trigka@ceid.upatras.gr

Abstract: The evolution of smart cities is intrinsically linked to advancements in computing paradigms that support real-time data processing, intelligent decision-making, and efficient resource utilization. Edge and cloud computing have emerged as fundamental pillars that enable scalable, distributed, and latency-aware services in urban environments. Cloud computing provides extensive computational capabilities and centralized data storage, whereas edge computing ensures localized processing to mitigate network congestion and latency. This survey presents an in-depth analysis of the integration of edge and cloud computing in smart cities, highlighting architectural frameworks, enabling technologies, application domains, and key research challenges. The study examines resource allocation strategies, real-time analytics, and security considerations, emphasizing the synergies and trade-offs between cloud and edge computing paradigms. The present survey also notes future directions that address critical challenges, paving the way for sustainable and intelligent urban development.

Keywords: edge; cloud computing; smart cities; architectural models

check for updates

Academic Editors: Stefano Rinaldi and Alan Oliveira De Sá

Received: 11 February 2025
Revised: 24 February 2025
Accepted: 28 February 2025
Published: 6 March 2025

Citation: Trigka, M.; Dritsas, E. Edge and Cloud Computing in Smart Cities. *Future Internet* **2025**, *17*, 118. https://doi.org/10.3390/fi17030118

1. Introduction

The rapid expansion of data-intensive applications has necessitated a shift from cloud-centric architectures to integrated edge–cloud computing, addressing limitations in latency, bandwidth, and real-time processing. Edge computing decentralizes computation, enabling localized processing, while cloud computing provides large-scale analytics and long-term data storage. Their synergy ensures adaptive resource allocation, dynamic service provisioning, and intelligent workload migration across diverse applications [1,2].

Architectural advancements, including multi-tier hierarchies, fully distributed models, federated intelligence, and digital twin-enabled infrastructures, optimize performance for different operational needs. Hierarchical models enhance structured load balancing, while federated and distributed approaches prioritize privacy and decentralized intelligence. Artificial intelligence (AI)-driven automation, 5G/6G networks, and blockchain security further enable efficient orchestration, ultra-low-latency communication, and decentralized trust mechanisms [3–5].

The impact spans multiple domains, including smart cities, healthcare, transportation, industrial automation, and immersive applications. Edge-assisted AI enables real-time diagnostics, predictive maintenance, autonomous decision-making, and ultra-responsive Augmented Reality (AR)/Virtual Reality (VR) experiences, enhancing efficiency and sustainability. However, challenges remain in interoperability, workload migration, energy efficiency, security, and privacy. Heterogeneous edge devices, dynamic mobility, and cyberthreats require advanced AI models, security frameworks, and energy-aware computing strategies to ensure robust performance [6,7].

The present survey is motivated by the pressing need for advanced computing paradigms that can efficiently support the growing complexity of smart cities, where real-time data processing, intelligent decision-making, and dynamic resource optimization are critical. As urban environments become increasingly data-driven, traditional cloud-centric architectures face significant challenges in latency, bandwidth constraints, and real-time responsiveness. This survey addresses these limitations by exploring the synergy between edge and cloud computing, demonstrating how their integration can enhance latency-aware services, security, scalability, and adaptive intelligence in smart city applications.

A key contribution of this work, reflected in Table 1, is a comparative analysis of existing surveys, revealing critical gaps and overlooked aspects. Unlike prior studies that primarily focus on either edge computing or cloud computing in isolation, this survey provides a multi-tier architectural perspective, integrating AI-driven resource management, federated learning (FL), and security mechanisms to enable more efficient and autonomous smart-city ecosystems. Furthermore, while earlier works have predominantly reviewed existing frameworks and methodologies, this study distinguishes itself by offering a forward-looking perspective, examining the implications of emerging technologies such as 6G networks, quantum computing, and sustainable edge–cloud ecosystems. By addressing critical challenges in real-time processing, scalability, and cross-domain service orchestration, this survey establishes itself as a comprehensive and future-ready reference, reinforcing its novelty and significance in the field. In summary, this survey

- Provides an in-depth examination of multi-tier, fully distributed, FL-enhanced, and hybrid digital twin-enabled architectures, highlighting their scalability, resilience, and efficiency trade-offs.
- Explores the role of AI-driven resource allocation, 5G/6G networking, blockchain security, and federated intelligence in enhancing the performance, security, and privacy of edge–cloud infrastructures.
- Systematically assesses the impact of edge–cloud computing in smart transportation, healthcare, industrial automation, smart cities, energy management, AR/VR, disaster response, and cybersecurity, demonstrating its transformative potential.
- Highlights critical challenges outlining future research directions to address existing limitations.

Figure 1 presents a comprehensive overview of edge and cloud computing in smart cities, categorizing architectural models, enabling technologies, application domains, and emerging challenges. It depicts four key architectural frameworks: multi-tier hierarchical, fully distributed, clustered edge–cloud architecture with FL, and hybrid digital twin-enabled models, highlighting their strengths and trade-offs in latency, scalability, fault tolerance, and energy efficiency. The figure also maps enabling technologies such as 5G/6G networking, AI-driven resource allocation, blockchain security, and edge virtualization, illustrating their role in enhancing performance, security, and decentralized intelligence.

Furthermore, it outlines critical application domains, including smart healthcare, transportation, industrial automation, energy management, AR/VR, and cybersecurity, demonstrating the transformative potential of edge–cloud synergy. Lastly, future research directions such as AI-driven autonomy, federated intelligence, quantum computing, and sustainable edge–cloud ecosystems are highlighted, providing a roadmap for next-generation smart-city infrastructures.

Table 1. Summary of surveys on edge and cloud computing in smart cities.

Reference	Description	Focused Points	Limitations
[8]	Examines the development and implementation of smart cities, analyzing intelligent computing algorithms and their applications in urban environments. It provides insights into smart-city frameworks and various optimization techniques.	Covers smart-city frameworks and various optimization techniques.	Lacks deep insights into resource allocation strategies and AI-driven orchestration.
[9]	An overview of edge computing's role in smart cities, covering applications, classifications, and challenges. The paper also presents a taxonomy of edge computing applications for latency-sensitive smart-city services.	Focuses on latency-sensitive smart-city services.	Does not address integration challenges between edge and cloud computing.
[10]	Notes how cloud, mobile, and edge computing enhance smart cities by improving urban systems like health, energy, and planning. It highlights their role in addressing urban heat island effects and future integration challenges.	Explores the role of computing in urban planning, health, and energy management.	Limited discussion on real-time processing and AI-driven automation.
[11]	Discusses the advantages of edge computing in healthcare, the Internet of Things (IoT), and smart-city applications. Highlights edge computing's ability to enhance data security, reduce latency, and improve computational efficiency in real-time environments.	Emphasizes security, latency reduction, and computational efficiency.	Does not extensively discuss cloud–edge synergy.
[12]	Surveys the role of 5G-enabled multi-access edge computing (MEC) in smart cities. It highlights the potential of MEC to enhance smart-city infrastructure through reduced latency and distributed computing resources.	Focuses on how MEC improves smart-city infrastructure.	Does not compare MEC with other edge–cloud models.
[13]	Analyzes cloud computing security within smart-city networks, addressing threats, vulnerabilities, and countermeasures. The survey also discusses privacy concerns and the role of edge computing in mitigating security risks.	Focuses on security risks and privacy issues.	Limited focus on performance trade-offs and resource allocation.
This survey	Provides a comprehensive analysis of edge–cloud computing in smart cities, including architectures, resource allocation, AI integration, and security strategies.	- Offers a multi-tier architectural perspective. - Analyzes AI-driven resource allocation. - Compares security and privacy considerations. - Evaluates domain-specific applications (transportation, healthcare, etc.). - Outlines future research directions (6G, quantum, sustainable computing).	No major limitations compared to existing surveys, but future work may explore more real-world deployments and experimental results.

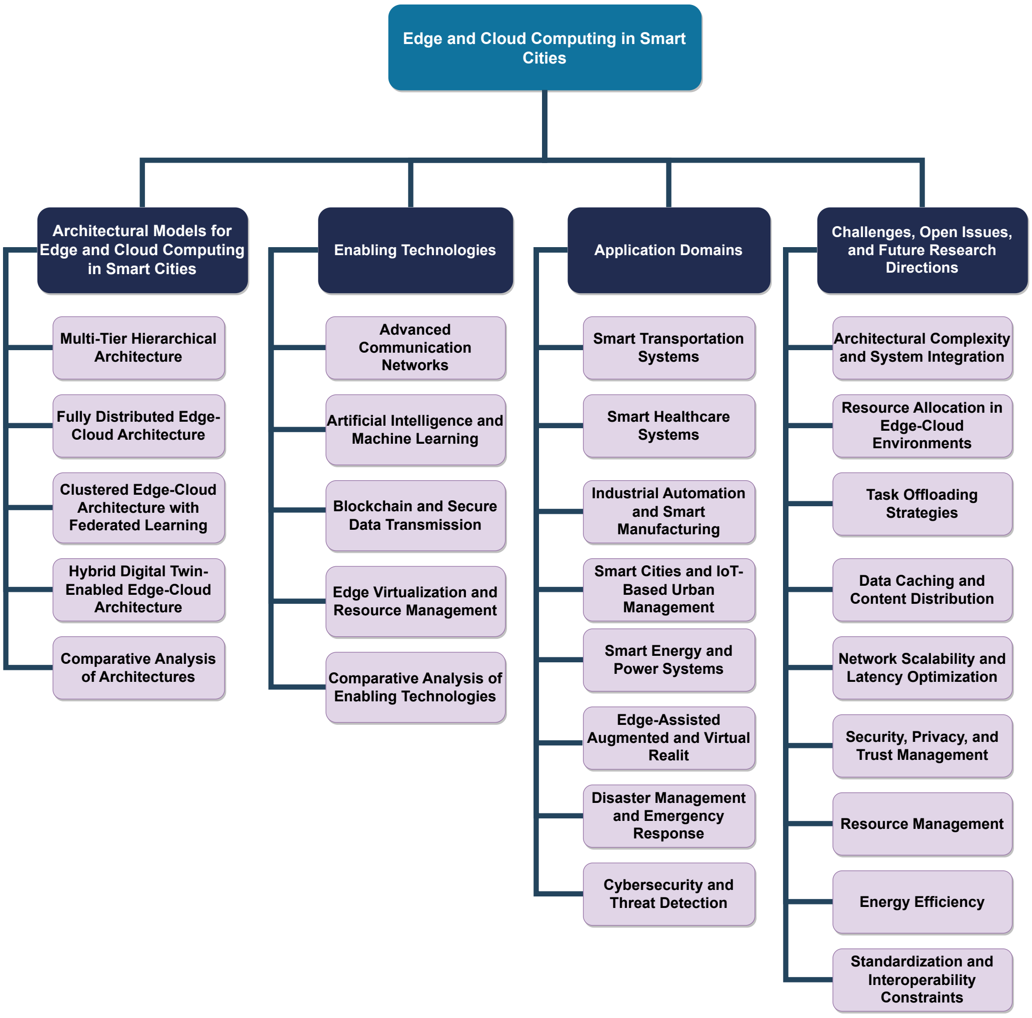

Figure 1. An overview of surveyed key topics: edge and cloud computing in smart cities.

More specifically, the remainder of the paper is structured as follows. Section 2 focuses on architectural models for edge and cloud computing in smart cities. Section 3 explores enabling technologies. Moreover, Section 4 notes applications domains. Section 5 provides challenges, open issues, and future research directions. Finally, Section 6 summarizes the findings of this survey.

2. Architectural Models for Edge and Cloud Computing in Smart Cities

The architectural models governing edge and cloud computing integration in smart cities define the distribution of computational tasks, data flow, and communication among various entities. These architectures address the trade-offs between latency, computational power, bandwidth consumption, and energy efficiency. A well-structured architectural framework is important to achieving optimal service delivery in applications requiring real-time decision-making and large-scale data analytics.

2.1. Multi-Tier Hierarchical Architecture

The multi-tier hierarchical architecture structures computational resources across multiple layers to optimize performance, latency, and resource utilization. This architecture balances centralized high-performance computing with distributed low-latency processing, ensuring intelligent service provisioning across urban environments [14,15].

Formally, the architecture is modeled as a layered graph $G = (\mathcal{N}, \mathcal{L})$, where $\mathcal{N}$ represents the set of computing nodes categorized into layers, and $\mathcal{L}$ denotes the communication links between them. Each node $n_i \in \mathcal{N}$ has a defined computational capacity, storage, and processing latency. The objective is to minimize cumulative service delay while optimizing resource allocation [16,17]. The computational hierarchy, shown in Figure 2, consists of three basic layers unwrapped into "Cloud, Edge, and Device" [18].

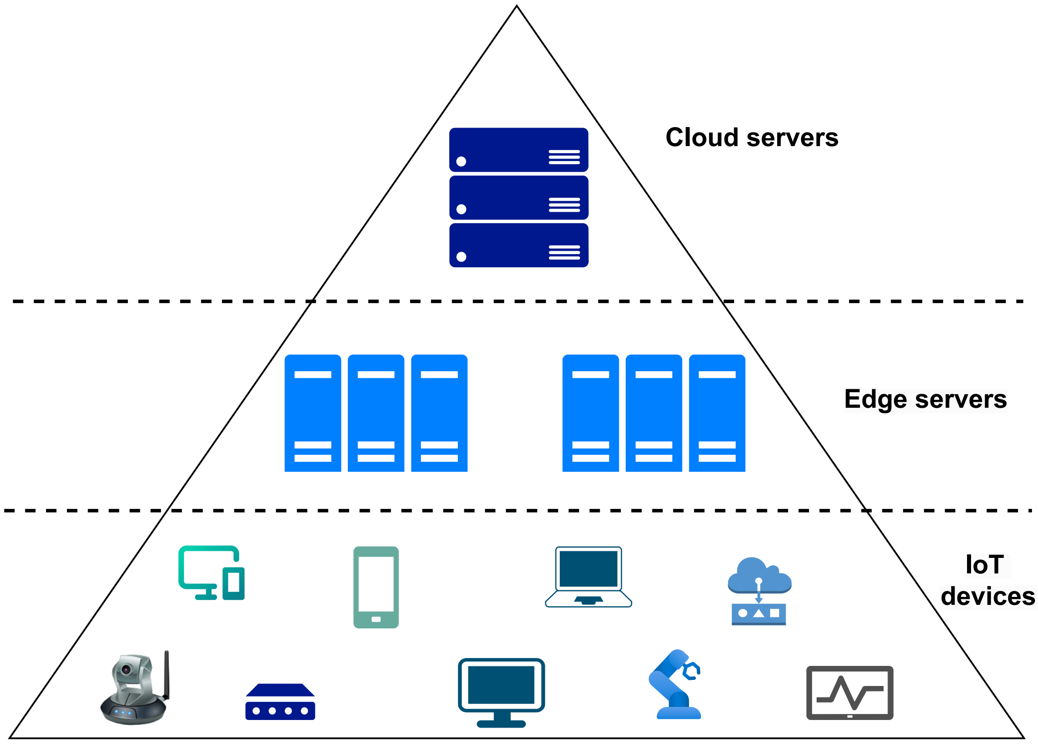

Figure 2. Schematic representation of the three-tier architecture.

Cloud layer C consists of M distinct nodes, the cloud servers, where each node provides centralized data storage, large-scale computational resources for data processing, AI model training, and global analytics. Computational resources at this layer are represented by

$$C = \{C_1, C_2, \dots, C_M\}, \quad C_m = (P_m, S_m, L_{C_m}), \quad m = 1, 2, \dots, M, \tag{1}$$

where P_m denotes the processing power of cloud node C_m, S_m its available storage, and L_{C_m} its inherent processing latency. The term L_{C_m} specifically represents the intrinsic processing delay at cloud node C_m, which depends solely on the node's computational capacity and workload. However, tasks offloaded to the cloud experience an additional transmission delay, leading to the total execution latency at the cloud, denoted as T_C. This total delay includes both network transmission time and processing latency in the cloud

$$T_C = T_{D \to C} + L_{C_m} + T_{C \to D}, \tag{2}$$

where $T_{D \to C}$ is the time required to transmit data from the device or edge to the cloud, L_{C_m} is the actual processing delay at the cloud node, and $T_{C \to D}$ is the response time for sending the processed results back to the device. The primary drawback of the cloud layer is the high transmission latency, T_C, which varies with network conditions and task size [19,20].

Edge layer E serves as an intermediary stratum that minimizes response times by processing tasks closer to data sources, alleviating network congestion and reducing latency. It is composed of K distinct nodes, called edge servers, each with its own computational and storage constraints. The edge layer is formally defined as

$$E = \{E_1, E_2, \ldots, E_K\}, \quad E_k = (P_k, S_k, L_{E_k}), \quad k = 1, 2, \ldots, K, \tag{3}$$

where P_k represents the processing power, S_k the storage capacity, and L_{E_k} the inherent latency at edge node E_k. The latency at the edge layer, L_{E_k}, is significantly lower than at the cloud since tasks are processed closer to their sources. However, the edge is constrained by finite computing resources, which limits its capacity to handle computationally intensive tasks [21].

The device layer (D) comprises a set of IoT devices and user equipment that continuously generate data streams but possess limited computational and storage capabilities. These devices, typically operating in dynamic environments, rely on higher layers for processing-intensive tasks. Assuming N devices, the device layer is formally defined as

$$D = \{D_1, D_2, \ldots, D_N\}, \quad D_n = (P_n, S_n, L_n), \tag{4}$$

where each device D_n is characterized by P_n processing capacity, constrained due to hardware and energy limitations, S_n available storage, primarily used for buffering and temporary data retention, and L_n inherent latency, influenced by local computation and network transmission delays. Given these constraints, IoT devices offload computationally demanding tasks to the Edge and Cloud Layers to optimize performance, reduce energy consumption, and enable real-time analytics [22].

Assuming a set of tasks $\mathcal{T}$, a task $\tau \in \mathcal{T}$ can be executed either at the edge or the cloud, with partial execution allowed. The optimization model for balancing task execution between the edge and cloud layers is formulated as

$$\min_{\alpha_i} \sum_{i=1}^{K} \Big(\alpha_i T_{E_i} + (1 - \alpha_i) T_C \Big), \tag{5}$$

where $\alpha_i \in [0, 1]$ (execution ratio) determines how much of task τ is executed at E_i, with the remaining portion offloaded to the cloud [23]. The term T_C used in this equation refers to the total execution time when the task is processed in the cloud, encompassing both the network transmission delay and cloud processing latency. In contrast, the edge processing delay T_{E_i} accounts only for the local execution time at edge node E_i, which is generally lower but subject to resource constraints.

To ensure balanced workload distribution, constraints are imposed on computational resources at both the edge and cloud layers, which prevent edge nodes from exceeding their capacity

$$\sum_{\tau \in \mathcal{T}} \alpha_i C_\tau \leq C_{E_i}, \quad \forall E_i \in E, \tag{6}$$

$$\sum_{\tau \in \mathcal{T}} (1 - \alpha_i) C_\tau \leq C_C. \tag{7}$$

The term C_{E_i} represents the total computational capacity available at edge node E_i, meaning that the total workload assigned to E_i cannot exceed this limit. This ensures that edge nodes are not overloaded, maintaining low-latency processing and preventing performance degradation. Similarly, C_C represents the total computational capacity available in the cloud. Since the cloud has significantly higher processing power, this constraint prevents excessive offloading that could lead to network congestion or increased response times. C_τ represents the computational demand of task τ. In these equations, α_i determines the fraction of task τ that is executed at edge node E_i, while $(1 - \alpha_i)$ represents the fraction offloaded to the cloud. If $\alpha_i = 1$, the task is fully executed at the edge, and no workload is sent to the cloud. Conversely, if $\alpha_i = 0$, the task is completely offloaded to the cloud.

When $0 < \alpha_i < 1$, the task is partially executed at the edge, with the remaining part offloaded to the cloud, ensuring a hybrid execution strategy. These constraints work together to distribute tasks efficiently, ensuring that edge computing resources are fully utilized without exceeding their processing limits while preventing the cloud from being overwhelmed by offloaded tasks.

Additionally, a latency-aware decision function determines the most suitable execution location

$$F(\tau) = \begin{cases} E_j, & \text{if } T_{E_j} + T_{D \to E_j} \leq T_C \quad \text{where } E_j \in E \\ C, & \text{otherwise.} \end{cases} \tag{8}$$

This decision function ensures that a task is executed at an edge node E_j if the sum of the edge execution time T_{E_j} and the transmission time from the device to the edge $T_{D \to E_j}$ is lower than the total cloud execution time T_C, which includes both network delay and cloud processing time. Otherwise, the task is offloaded to the cloud. This dynamic decision mechanism enables the efficient allocation of computational resources based on real-time network conditions and task execution requirements.

To capture the overall energy consumption at different layers, both active and idle states are considered

$$E_{total} = \sum_{i=1}^{K} \left(P_{D \to E_i} T_{D \to E_i} + P_{E_i}^{\text{active}} \alpha_i T_{E_i} + P_{E_i}^{\text{idle}} (1 - \alpha_i) T_{E_i} + P_{E_i \to C} T_{E_i \to C} + P_C T_C \right), \tag{9}$$

where $P_{D \to E_i}$ is the power consumed for data transmission from devices to the edge, $P_{E_i}^{\text{active}}$ is the power used when processing at the edge, $P_{E_i}^{\text{idle}}$ accounts for background energy usage when idle, and $P_{E_i \to C}$ and P_C represent power usage for cloud communication and processing. This model ensures that idle energy consumption is accounted for, making it more realistic for power-constrained edge devices [24–26].

This hierarchical architecture efficiently balances latency, computational demand, and energy efficiency, enabling real-time smart-city applications. However, challenges such as network congestion, resource synchronization, and adaptive task migration require advanced AI-based scheduling techniques to enhance system resilience [27].

2.2. Fully Distributed Edge–Cloud Architecture

The fully distributed edge–cloud architecture eliminates the constraints of hierarchical computing models by enabling decentralized processing, decision-making, and coordination among edge nodes and cloud resources. In contrast to traditional architectures where task execution follows a predefined hierarchy, this model ensures dynamic workload distribution across edge nodes, reducing bottlenecks and improving scalability and fault tolerance [28,29].

A fully distributed edge–cloud system can be represented by an undirected graph $G(\mathcal{N}, \mathcal{L})$ where $\mathcal{N} = E \cup C$ is the set of computational nodes, consisting of edge nodes E and cloud nodes C. The set $\mathcal{L}$ represents bidirectional communication links between these nodes, enabling distributed decision-making and resource sharing [30,31].

Unlike hierarchical models where cloud resources predominantly determine task execution, the distributed model allows edge nodes to autonomously decide whether to process a task locally or offload it to a neighboring node or cloud resource. For each task τ arriving at edge node $E_j \in E$, the execution decision is determined based on the following latency-aware rules function

$$F(\tau) = \begin{cases} E_j, & \text{if } T_{E_j} \leq T_C \text{ and } T_{E_j} \leq T_{E_k} + T_{\text{comm}}(E_j, E_k), \\ E_k, & \text{if } T_{E_k} + T_{\text{comm}}(E_j, E_k) \leq T_{E_j} \text{ and } T_{E_k} + T_{\text{comm}}(E_j, E_k) \leq T_C, \\ C, & \text{otherwise.} \end{cases} \tag{10}$$

where T_{E_j}, T_{E_k} is the task execution time at edge node E_j and E_k, respectively, T_C is the execution time at the cloud, and in case of task offloading to a neighboring node, $T_{\text{comm}}(E_j, E_k)$ denotes the communication delay between nodes j and k. This decision function aims to minimize execution latency while accounting for network constraints and computational limitations [32,33]. The function prioritizes local execution at E_j if it offers the lowest latency. If a neighboring edge node E_k executes the task faster than E_j and the cloud, the task is migrated to E_k. If neither local execution nor edge-to-edge migration is feasible, the task is offloaded to the cloud.

Each edge node has finite computational capacity C_{E_j} and energy budget $E_{E_j}^{budget}$. The workload assigned to an edge node at any time is constrained by

$$\sum_{\tau \in \mathcal{T}_{E_j}} C_\tau \leq C_{E_j}, \sum_{\tau \in \mathcal{T}_{E_j}} E_\tau^{cons} \leq E_{E_j}^{budget}, \quad \forall E_j \in E, \tag{11}$$

where $\mathcal{T}_{E_j}$ represents the set of tasks assigned to node E_j (which is a subset of the total tasks $\mathcal{T}$), and C_τ and E_τ^{cons} denote the computational demand and energy consumption of task τ, respectively [34,35].

In the distributed model, edge nodes collaborate dynamically to balance workload distribution. The probability of task offloading from one edge node to another is governed by

$$P_{\text{offload}}(E_j \to E_k) = \frac{1}{1 + e^{-\lambda(\theta_k - \theta_j)}}, \tag{12}$$

where θ_j and θ_k denote the available computational capacity of nodes E_j and E_k, respectively, and λ is a sensitivity parameter controlling the offloading decision. If an edge node's available capacity falls below a threshold θth, it attempts to offload tasks to neighboring nodes before considering cloud offloading [36,37].

The total system latency T_{sys} in a fully distributed edge–cloud network is expressed as

$$T_{\text{sys}} = \sum_{\tau \in \mathcal{T}} \left(T_{E_j} + \frac{d_{E_j,E_k}}{B_{E_j,E_k}} + T_C \delta_\tau \right), \tag{13}$$

where d_{E_j,E_k} represents the distance between two edge nodes, B_{E_j,E_k} is the available bandwidth for communication, and δ_τ is an indicator function, where $\delta_\tau = 1$ if the task is processed in the cloud, and $\delta_\tau = 0$ if processed at the edge [38,39].

The distributed nature of this architecture enhances fault tolerance. If an edge node E_j fails, its workload is reallocated to neighboring nodes without interrupting service. The failure probability of a task execution in this model is given by

$$P_f = 1 - \prod_{j=1}^{K} (1 - p_{E_j}), \tag{14}$$

where p_{E_j} represents the failure probability of node E_j. As the number of cooperative nodes increases, the probability of successful execution improves [40,41].

Energy efficiency is a crucial consideration in fully distributed architectures, especially where edge nodes operate on limited power. The total energy consumption of the system is given by

$$E_{\text{total}} = \sum_{j=1}^{K} \Big(P_{E_j} T_{E_j} + P_{\text{comm}} T_{\text{comm}}(E_j, E_k) + P_C T_C \delta_\tau \Big), \tag{15}$$

where P_{E_j} and P_{comm} are the power consumption rates for computation and communication, respectively. Efficient scheduling strategies, such as reinforcement learning (RL)-based task allocation, can optimize energy efficiency by dynamically adjusting resource usage [42,43].

The fully distributed edge–cloud architecture removes the limitations of hierarchical computing, ensuring adaptability to variations in workload, network congestion, and node availability. This cooperative processing framework maximizes resource utilization and minimizes latency while maintaining service reliability. However, the model introduces challenges such as increased synchronization complexity and the need for consensus mechanisms to maintain consistency across edge nodes [44,45].

2.3. Clustered Edge–Cloud Architecture with Federated Learning

The clustered edge–cloud architecture with FL introduces a structured, decentralized approach to computational resource allocation, where edge nodes are grouped into dynamically coordinated clusters that collaborate with cloud servers. This architectural model optimizes computational efficiency by minimizing latency, reducing network congestion, and preserving data privacy, thereby enhancing the scalability and adaptability of edge computing in smart-city environments. Unlike fully distributed architectures, which operate without a predefined structure, the clustered model ensures systematic coordination, enabling intelligent workload distribution and federated machine learning (ML) while mitigating inter-cluster communication overhead [46,47].

The architecture consists of a set of clusters

$$C_E = \{Cl_1, Cl_2, \ldots, Cl_Q\}, \tag{16}$$

where each cluster Cl_q comprises multiple edge nodes $\mathcal{E}_q$ and a cluster coordinator G_q. Each cluster is dynamically formed based on spatial proximity, resource availability, and computational demand. The set of N_q edge nodes within the qth cluster is denoted as

$$E_q = \{E_{q1}, E_{q2}, \ldots, E_{qN_q}\}, \quad q = \{1, 2, \ldots, Q\}, \tag{17}$$

where each edge node E_{qi} $(i = 1, 2, \ldots, N_q)$ is characterized by a triple (P_{qi}, S_{qi}, L_{qi}), representing processing power, available storage, and inherent latency, respectively. The cluster coordinator G_q manages intra-cluster task distribution and FL aggregation. The selection of a cluster coordinator follows an optimization criterion

$$G_q = \arg\min_{E_{qi} \in \mathcal{E}_q} \Big(\alpha L_{qi} - \beta P_{qi} - \gamma S_{qi} \Big), \tag{18}$$

where α, β, γ are weighting coefficients balancing latency, computational power, and storage capacity in selecting the optimal coordinator. Higher processing power and storage are preferred, ensuring efficient coordination while minimizing latency [48–50].

Each cluster follows a hierarchical processing framework, where tasks are first assigned to an available edge node within the cluster based on the optimization function

$$E^*_{qi} = \arg\min_{E_{qi} \in \mathcal{E}_q} \left(T_{E_{qi}} + \frac{d_{E_{qi}, G_q}}{B_{E_{qi}, G_q}} \right), \tag{19}$$

where $T_{E_{qi}}$ represents the processing delay at edge node E_{qj}, d_{E_{qi}, G_q} denotes the communication distance to the cluster coordinator, and B_{E_{qi}, G_q} is the available bandwidth between E_{qi}

and G_q. If no edge node within the cluster meets the latency constraint, the task is offloaded to the cloud according to

$$F(\tau) = \begin{cases} E_{qi}^*, & \text{if } T_{E_{qi}^*} + T_{\text{comm}}(E_{qi}, G_q) \leq T_{\text{th}}, \text{where } E_{qi} \in E_q \\ C, & \text{otherwise}, \end{cases} \tag{20}$$

where T_{th} is the maximum latency threshold for real-time processing [51,52].

FL is integrated into this architecture to facilitate collaborative model training without exposing raw data to external networks. Each edge node E_{qi} maintains a local ML model $\mathcal{M}_i$ and updates it using local datasets Dt_i following

$$\mathcal{M}_i^t = \mathcal{M}_i^{t-1} - \eta \nabla L(\mathcal{M}_i^{t-1}, Dt_i), \tag{21}$$

where η is the learning rate, and ∇L represents the gradient of the loss function. The locally trained models are then transmitted to the cluster coordinator for aggregation

$$\mathcal{M}_q^t = \sum_{i=1}^{N_q} w_i \mathcal{M}_i^t, \quad \sum_{i=1}^{N_q} w_i = 1, \tag{22}$$

where w_i represents the weight assigned to each edge node based on its dataset size. The aggregated model is periodically synchronized with the global cloud model

$$\mathcal{M}_C^t = \sum_{q=1}^{Q} v_q \mathcal{M}_q^t, \quad \sum_{q=1}^{Q} v_q = 1, \tag{23}$$

where v_q is the aggregation weight assigned to each cluster. This FL mechanism ensures privacy preservation, reduces cloud communication costs, and enhances model adaptability to local conditions [53,54].

The overall system latency, encompassing computational, communication, and learning synchronization delays, is expressed as

$$T_{\text{sys}} = \sum_{\tau \in \mathcal{T}} \Big(T_{E_{qi}} + T_{\text{comm}}(E_{qi}, G_q) + T_{\text{agg}} + T_C \delta_\tau \Big), \tag{24}$$

where T_{agg} denotes the time required for model aggregation at the cluster level, and T_C captures additional delays if cloud interaction is required [55].

A key advantage of the clustered architecture lies in its fault tolerance and resilience to node failures. If an edge node becomes unavailable, its computational workload is dynamically reassigned to neighboring nodes within the same cluster. The probability of complete failure is determined by assessing the likelihood that all nodes in a cluster q fail simultaneously. The following equation aggregates individual node failure probabilities to evaluate system resilience and fault tolerance

$$P_f = 1 - \prod_{i=1}^{N_q} (1 - p_{E_{qi}}), \tag{25}$$

where $p_{E_{qi}}$ is the failure probability of node E_{qi}. The probability of service continuity increases with the number of edge nodes in the cluster, ensuring robustness in decentralized environments [56].

Energy efficiency in this architecture is enhanced by restricting cloud interactions and optimizing intra-cluster task execution. The total energy consumption across clusters is given by:

$$E_{\text{total}} = \sum_{i=1}^{N_q} \Big(P_{E_{qi}} T_{E_{qi}} + P_{\text{comm}} T_{\text{comm}}(E_{qi}, G_q) + P_{\text{agg}} T_{\text{agg}} + P_C T_C \delta_\tau \Big), \tag{26}$$

where P_{agg} represents the power consumption associated with FL model aggregation. Adaptive power management strategies, such as dynamic voltage scaling and sleep scheduling, further improve energy efficiency [57,58].

2.4. Hybrid Digital Twin-Enabled Edge–Cloud Architecture

The hybrid digital twin-enabled edge–cloud architecture integrates computational capabilities across distributed edge nodes and centralized cloud resources while incorporating real-time virtual representations of physical systems. By leveraging digital twins, this architecture enhances predictive analytics, adaptive decision-making, and dynamic optimization of urban environments, enabling intelligent automation and real-time monitoring. The architecture is structured to maintain seamless interactions between the physical system, its digital counterpart, and the computational infrastructure, ensuring data consistency and low-latency execution [59,60].

The computational framework is modeled as a set of interconnected layers, where each physical entity in the smart-city environment is associated with a digital twin. This system is formalized as

$$G = (\mathcal{P}, \mathcal{D}, E, C, \mathcal{L}), \tag{27}$$

where $\mathcal{P}$ represents the set of physical entities, $\mathcal{D}$ denotes the digital-twin models corresponding to each entity, E consists of the edge nodes responsible for localized processing, C includes cloud resources performing large-scale analytics, and $\mathcal{L}$ defines the set of communication links interconnecting these components [61,62].

The digital twin $\mathcal{D}_i$ associated with a physical entity $\mathcal{P}_i$ maintains a continuous state synchronization mechanism to ensure accurate real-time representation. The update cycle follows

$$\mathcal{D}_i^{t+1} = f(\mathcal{D}_i^t, \mathcal{S}_i, \delta t), \tag{28}$$

where $\mathcal{D}_i^{t+1}$ represents the updated digital-twin state at time $t+1$, $\mathcal{S}_i$ denotes the set of sensor inputs from $\mathcal{P}_i$, and δt is the time step governing synchronization frequency. The function f encapsulates the transformation of sensor data into a virtual model, ensuring consistency with the real-world entity [63,64].

The decision-making process in this architecture is governed by an optimization function that determines the optimal execution layer for each computational task τ. The execution strategy follows

$$F(\tau) = \begin{cases} E_j, & \text{if } T_{E_j} + T_{sync} \leq T_C + T_{comm}, \text{where } E_j \in E \\ C, & \text{otherwise,} \end{cases} \tag{29}$$

where T_{E_j} is the processing delay at the edge node E_j, T_{sync} denotes the synchronization delay between the digital twin and the physical entity, T_C represents the cloud processing delay, and T_{comm} is the communication latency between the edge and cloud. The selection function prioritizes execution at the edge layer unless cloud processing becomes necessary due to resource constraints or computational complexity [65,66].

Synchronization latency in digital-twin architectures significantly impacts real-time system performance. The total synchronization delay is modeled as

$$T_{sync} = \frac{d_{update} + d_{comm} + d_{proc}}{f_{sync}}, \tag{30}$$

where d_{update} represents the time taken for sensor data acquisition, d_{comm} captures the transmission delay from the physical system to the digital twin, d_{proc} accounts for the processing time required to update the twin's state, and f_{sync} is the synchronization frequency. The objective is to minimize T_{sync} to ensure real-time consistency between the physical and digital environments [67,68].

Computational load balancing between the edge and cloud layers is a fundamental aspect of this architecture, as resource allocation must dynamically adapt to real-time conditions. The total workload across the architecture is represented by

$$\sum_{j=1}^{K} a_j C_{E_j} + (1 - a_j) C_C = C_{total}, \tag{31}$$

where a_j is a binary variable indicating whether task τ is processed at the edge ($a_j = 1$) or offloaded to the cloud ($a_j = 0$), C_{E_j} denotes the computational capacity of the edge node, C_C represents the cloud's processing capability, and C_{total} is the overall system workload [69,70].

In addition to computational efficiency, energy consumption remains a critical factor in determining the viability of digital twin-enabled architectures. The total energy consumption is given by:

$$E_{total} = \sum_{j=1}^{K} \left(P_{E_j} T_{E_j} + P_{sync} T_{sync} + P_C T_C \right), \tag{32}$$

where P_{E_j} represents the power consumption of edge node E_j, P_{sync} accounts for the energy required to maintain synchronization, and P_C denotes the power cost of cloud-based processing. Efficient task scheduling algorithms, such as RL-based optimizations, can be incorporated to minimize E_{total} while maintaining system performance [71,72].

The hybrid edge–cloud model offers several advantages over conventional architectures by integrating real-time simulation, predictive analytics, and adaptive resource allocation. However, maintaining consistency between the digital twin models and their physical counterparts introduces computational overhead, particularly in high-frequency synchronization scenarios. To address this challenge, adaptive synchronization strategies dynamically adjust f_{sync} based on task urgency and network conditions, ensuring efficient data transmission while preventing excessive update cycles [73–75].

From a fault-tolerance perspective, system resilience is achieved through distributed redundancy mechanisms. In the event of an edge node failure, the system dynamically redistributes computational tasks and synchronization responsibilities to adjacent nodes, mitigating service disruptions. The probability of failure in this architecture is expressed as

$$P_f = 1 - \prod_{j=1}^{K} (1 - p_{E_j}), \tag{33}$$

where p_{E_j} denotes the failure probability of an individual edge node, and K represents the total number of edge nodes supporting redundancy. By increasing the number of participating nodes, the likelihood of service continuity is enhanced [76,77].

2.5. Comparative Analysis of Architectures

The integration of edge and cloud computing within smart-city infrastructures follows distinct architectural paradigms, each addressing key challenges such as latency, scalability, fault tolerance, energy efficiency, and computational complexity. Selecting the right architecture significantly impacts system performance, resource utilization, and efficiency, making a comparative analysis essential for real-time and large-scale applications.

Latency is critical for responsiveness in edge–cloud systems. Hierarchical architectures involve multiple layers, which can introduce delays due to transmission overhead. Fully distributed models minimize latency by processing tasks closer to data sources, reducing cloud dependency. Clustered architectures improve efficiency by structuring edge nodes into localized groups, ensuring faster response times. Meanwhile, digital twin-enabled architectures may experience additional delays due to the need for continuous synchronization, impacting ultra-low-latency applications.

Scalability defines how well an architecture can handle increasing workloads. Hierarchical models rely on cloud computing, which scales vertically but can face congestion issues. Fully distributed approaches support horizontal scaling by dynamically reallocating tasks across edge nodes, improving adaptability. Clustered architectures optimize local resource management, offering a balance between cloud-based and edge-based scalability. Digital-twin architectures enhance system adaptability by simulating resource demands, allowing preemptive adjustments.

Fault tolerance ensures system reliability despite node failures. Hierarchical architectures are more vulnerable due to their reliance on cloud infrastructure. Fully distributed models enhance resilience through cooperative processing, ensuring that failures at one node do not disrupt overall operations. Clustered architectures provide moderate fault tolerance by redistributing workloads within each cluster. Digital twin-enabled models further improve resilience by maintaining virtual representations of physical components, allowing proactive failure mitigation.

Energy efficiency is crucial for smart-city applications, especially in power-constrained environments. Fully distributed models typically consume less energy by processing data locally, reducing transmission costs. Hierarchical models involve frequent cloud communication, increasing energy usage. Clustered architectures balance energy consumption by minimizing long-range data transfers. Digital twin-based architectures, while enhancing system intelligence, may lead to higher energy consumption due to continuous synchronization and processing.

Computational complexity varies across architectures. Hierarchical models follow structured workflows, making them relatively simple to implement but less flexible. Fully distributed systems introduce higher complexity due to decentralized decision-making and real-time task balancing. Clustered models mitigate this by organizing edge resources into manageable units, reducing system-wide overhead. Digital-twin architectures, though highly adaptive, require extensive real-time processing, making them computationally intensive.

Each architecture presents unique trade-offs based on latency, scalability, fault tolerance, energy efficiency, and complexity. Hierarchical architectures offer structured workload distribution but struggle with scalability and fault tolerance. Fully distributed models maximize resilience but require advanced coordination. Clustered architectures provide a balance between scalability and efficiency. Digital-twin architectures enhance predictive decision-making but introduce synchronization overhead. The choice of architecture should align with application-specific requirements, ensuring optimal system performance. In summary, Table 2 presents a comparative analysis of the different architectures based on key performance metrics.

Table 2. Comparative Analysis of Edge-Cloud Architectures.

Architecture	Latency	Scalability	Fault Tolerance	Energy Efficiency	Computational Complexity
Hierarchical [14–27]	Moderate. Multi-layer processing increases latency but improves structured workload allocation.	Moderate–High. It can scale by adding cloud resources but is constrained by layer dependencies.	Low. Cloud dependency creates a single point of failure, reducing reliability.	Moderate. Edge reduces energy consumption, but inter-layer communication overhead remains.	Low. Task allocation follows predefined deterministic execution models.
Fully distributed [28–45]	Low. Decentralized execution reduces transmission delays, improving responsiveness.	High. Adaptive task allocation enables horizontal scalability without reliance on the cloud.	High. Redundant nodes allow task redistribution, ensuring minimal service disruption.	High. Execution at the edge reduces data transmission energy costs.	High. Requires real-time synchronization and decentralized scheduling strategies.
Clustered edge–cloud [46–58]	Low–Moderate. Clusters handle local processing, but cloud involvement adds minimal delay.	High. Cluster controllers optimize workload balancing across multiple nodes.	Moderate–High. Node failures are managed within clusters, but controller failures impact performance.	Moderate–High. Local execution is efficient, but cloud synchronization increases overhead.	Moderate. Cluster-level processing improves efficiency while reducing global complexity.
Hybrid digital twin-enabled [59–77]	Moderate–High. Digital-twin synchronization introduces additional processing delay.	High. Supports predictive analytics for proactive system scaling.	High. Digital twins maintain a system state even when physical components fail.	Moderate. Frequent updates impact power efficiency; reducing synchronization frequency mitigates this.	High. Requires continuous real-time data processing and AI-driven analytics.

3. Enabling Technologies

The integration of edge and cloud computing in smart cities relies on a set of enabling technologies that enhance computational efficiency, network performance, data security, and intelligent automation. These technologies form the backbone of modern computing infrastructures, allowing seamless interactions between distributed processing units and centralized cloud resources. The interplay among advanced communication protocols, artificial intelligence-driven optimizations, and secure data transmission mechanisms dictates the efficiency and scalability of edge–cloud architectures. This section presents a detailed analysis of the key enabling technologies, highlighting their mathematical formulations and impact on system performance.

3.1. Advanced Communication Networks

Effective communication networks are crucial for enabling seamless interaction between edge nodes, cloud resources, and end-user devices in smart cities. The performance of edge–cloud computing systems depends on low-latency, high-bandwidth connectivity to support real-time applications [78–80].

First, 5G and 6G technologies play a pivotal role in improving network efficiency by providing ultra-reliable low-latency communication (URLLC), massive machine-type communication (mMTC), and enhanced mobile broadband (eMBB). These advancements ensure that smart-city applications can handle vast amounts of data with minimal transmission delays [81].

To further enhance network efficiency, Software-Defined Networking (SDN) and Network Function Virtualization (NFV) enable dynamic network configuration and resource optimization. SDN decouples the control and data planes, allowing for flexible network management, while NFV virtualizes network functions, reducing hardware dependencies and improving scalability [82–84].

Another critical factor is interference management, which affects signal quality and data throughput. High interference levels can cause network congestion and delays, impacting real-time applications. By implementing intelligent traffic routing and adaptive bandwidth allocation, communication networks can mitigate interference issues, ensuring reliable data exchange across edge–cloud infrastructures [85,86].

Overall, advanced communication technologies facilitate efficient data transmission in edge–cloud environments, making them fundamental to the success of latency-sensitive smart-city applications.

3.2. Artificial Intelligence and Machine Learning

The integration of AI and ML in edge–cloud computing enhances decision-making, resource management, and predictive analytics. AI-driven techniques help optimize task scheduling, improve computational efficiency, and ensure adaptive service provisioning.

One of the key applications of AI in edge–cloud computing is intelligent workload distribution. AI models analyze real-time conditions, such as network latency, processing capacity, and energy consumption, to determine whether a task should be executed at the edge or offloaded to the cloud. This dynamic allocation minimizes response times and optimizes resource utilization [87–89].

Another important aspect is FL, which allows edge devices to collaboratively train AI models without sharing raw data. Instead of sending complete datasets to a central server, FL enables decentralized model updates, preserving data privacy while improving overall system intelligence. This approach is particularly useful in healthcare, transportation, and other sensitive domains where data confidentiality is a priority [90,91]. Additionally, RL techniques are used to adapt resource allocation strategies over time. By continuously learning from system performance, RL-based models can dynamically adjust processing power, bandwidth allocation, and task prioritization, ensuring efficient edge–cloud operations [92]. AI and ML significantly enhance the scalability and responsiveness of edge–cloud systems, enabling smarter, more adaptive computing frameworks in smart-city applications.

3.3. Blockchain and Secure Data Transmission

Security is a critical concern in edge–cloud computing, where large-scale data transmission and processing occur across multiple distributed nodes. Blockchain technology enhances security by providing a decentralized framework that ensures data integrity, transparency, and protection against unauthorized modifications. A key advantage of blockchain is its ability to create tamper-proof transaction records. In edge–cloud environments, blockchain secures communication between edge nodes and the cloud by verifying each transaction through a consensus mechanism. This prevents malicious entities from altering data and strengthens trust among interconnected devices [93,94].

Another essential aspect of secure data transmission is end-to-end encryption. Advanced cryptographic techniques, such as Elliptic Curve Cryptography (ECC) and Zero-

Trust Security Frameworks, ensure that only authorized entities can access sensitive information. Unlike traditional security models that assume trust within a network, the zero-trust approach continuously verifies identities and access permissions, reducing the risk of cyberthreats [95,96].

Furthermore, multi-factor authentication (MFA) and intrusion detection systems (IDS) enhance cybersecurity by preventing unauthorized access and identifying anomalies in network activity. These mechanisms help mitigate threats such as data breaches, denial-of-service (DoS) attacks, and unauthorized system modifications [97,98].

By integrating blockchain with encryption techniques and zero-trust models, edge–cloud computing can achieve enhanced security, data integrity, and resilience against cyberattacks, ensuring the safe deployment of smart-city services.

3.4. Edge Virtualization and Resource Management

Virtualization technologies are essential for managing computational resources efficiently in edge–cloud environments. By enabling multiple applications to share processing infrastructure dynamically, virtualization enhances scalability, flexibility, and cost-effectiveness.

One of the key benefits of virtualization is containerization, which allows applications to run in isolated environments with minimal overhead. Containers provide a lightweight alternative to virtual machines (VMs), reducing the complexity of deploying and managing workloads at the edge. This approach is widely used in microservice-based architectures, where applications are broken down into smaller, modular components [99–101].

Another important aspect of resource management is dynamic workload scaling. Edge–cloud systems must adjust computational resources based on real-time demand to maintain optimal performance. When the workload increases, additional virtual instances can be deployed to handle the demand. Conversely, during low-traffic periods, resources can be deallocated to save energy [102].

To further enhance efficiency, energy-aware resource management strategies are implemented. These include techniques such as dynamic voltage and frequency scaling (DVFS), which adjusts processing power based on workload intensity to reduce energy consumption. By optimizing power usage, edge–cloud systems can achieve sustainability without compromising performance [103].

Effective virtualization and intelligent resource management enable seamless workload distribution, energy-efficient computing, and adaptive service provisioning, making them fundamental for large-scale edge–cloud infrastructures in smart cities [104,105].

3.5. Comparative Analysis of Enabling Technologies

The integration of enabling technologies within edge–cloud computing frameworks enhances efficiency, scalability, and resilience by optimizing latency, computational intelligence, security, and resource allocation. Their effectiveness depends on improving system performance while minimizing energy consumption, response time, and computational overhead. This section compares their contributions and trade-offs.

High-speed communication protocols, such as 5G, 6G, and Wi-Fi 6, reduce latency and enhance data transmission rates, improving real-time interactions. While 5G supports ultra-reliable low-latency communication (URLLC), its infrastructure costs remain high. Future 6G networks promise lower latency but introduce higher power consumption and signal stability challenges over long distances.

AI-driven resource management improves task scheduling and workload balancing through RL and FL. RL dynamically adjusts resource allocation, while FL decentralizes model training to enhance privacy. However, FL introduces synchronization delays and additional communication costs, requiring optimized coordination.

Security mechanisms, particularly blockchain-based authentication, mitigate unauthorized access risks in decentralized edge–cloud environments. Traditional mechanisms impose high computational costs, making them less viable for real-time applications. Lightweight alternatives, incorporating ECC and zero-trust models, enhance security while minimizing overhead.

Virtualization technologies, including containerization, facilitate dynamic resource allocation and multi-tenancy. Containers provide faster deployment and lower overhead than VMs, making them well suited for edge workloads. However, security concerns, particularly kernel vulnerabilities, necessitate robust isolation mechanisms.

Energy efficiency is a major challenge in edge–cloud computing, requiring a balance between performance and power consumption. Dynamic Voltage Scaling (DVS) and adaptive workload migration help optimize energy use by reallocating tasks to nodes with higher efficiency. While these techniques improve sustainability, they require accurate predictive models to prevent performance degradation.

Each enabling technology plays a distinct role in optimizing edge–cloud infrastructures. High-speed networks improve latency-sensitive applications, AI-driven optimization enhances resource management, blockchain strengthens security, and virtualization improves computational efficiency. The selection of an optimal technology combination depends on application-specific requirements, including latency constraints, security considerations, and computational trade-offs. A summarized comparative analysis is presented in Table 3, highlighting each technology's contributions, limitations, and trade-offs.

Table 3. Comparative analysis of enabling technologies in edge–cloud computing.

Technology	Functionality	Benefits	Limitations	Influence on Edge-Cloud Computing
Advanced communication networks [78–86]	High-speed data transmission, low-latency networking, real-time routing.	Enhances responsiveness, minimizes delays, maximizes throughput.	High deployment costs, spectrum allocation complexity, interference management.	Ensures fast and reliable connectivity between edge and cloud layers.
AI and ML [87–92]	Intelligent workload distribution, predictive analytics, real-time optimizations.	Enhances efficiency, automates decision-making, and reduces task execution time.	Computational overhead, real-time inference complexity, data privacy concerns.	Reduces latency, optimizes task execution, and improves adaptability in dynamic environments.
Blockchain and secure transmission [93–98]	Decentralized security, cryptographic authentication, integrity verification.	Ensures tamper-proof data transactions and eliminates reliance on centralized authorities.	High computational power demand, increased verification latency, and scalability challenges.	Strengthens trust and reliability in multi-node environments but introduces verification delays.
Edge virtualization and resource optimization [99–105]	Dynamic workload allocation, multi-tenant computing, containerized execution.	Improves system elasticity, enhances load balancing, and minimizes operational costs.	Complexity in orchestration, potential security vulnerabilities, resource contention.	Enables adaptive workload migration, optimizes resource distribution, and balances processing loads.

4. Application Domains

The deployment of edge and cloud computing architectures has revolutionized various application domains by enabling real-time data processing, intelligent decision-making, and efficient resource management. These architectures support diverse industries such as transportation, healthcare, industrial automation, and smart-city infrastructure while also transforming energy management, immersive AR/VR experiences, disaster response, and cybersecurity. By optimizing computational efficiency and reducing latency, edge–cloud frameworks ensure that mission-critical applications operate with seamless connectivity and adaptive intelligence. The convergence of distributed intelligence with cloud-based analytics fosters scalable and resilient ecosystems, addressing the growing complexities of modern digital infrastructures.

4.1. Smart Transportation Systems

The advancement of smart transportation relies on distributed intelligence for real-time traffic optimization, vehicle coordination, and safety management. Edge computing enables the local processing of vast streams of vehicular data, facilitating dynamic route adjustments, congestion mitigation, and intelligent traffic signal control. Cloud services aggregate large-scale mobility data, providing long-term analytics for infrastructure planning and predictive modeling [106,107].

Vehicle-to-everything (V2X) communication ensures seamless interaction between vehicles, roadside units, and cloud platforms, enabling low-latency data exchange crucial for collision avoidance and autonomous navigation. The increasing integration of AI enhances decision-making by predicting traffic patterns, optimizing resource allocation, and enabling cooperative driving strategies. While edge nodes process time-sensitive data for immediate action, cloud resources refine long-term mobility insights, ensuring a balance between real-time responsiveness and large-scale intelligence [108–110].

The deployment of autonomous vehicles intensifies the demand for ultra-reliable, low-latency processing. Edge-based AI algorithms support real-time sensor fusion and adaptive control, minimizing reliance on distant cloud servers. As mobility ecosystems become more interconnected, FL enables collaborative AI model training across distributed vehicle networks while preserving data privacy, improving real-time decision accuracy, and enhancing safety standards [111–113].

4.2. Smart Healthcare Systems

Edge–cloud computing has redefined healthcare through remote patient monitoring, intelligent diagnostics, and real-time emergency response. The proliferation of wearable medical devices and smart sensors enables continuous health tracking, where edge nodes analyze patient vitals and detect anomalies instantly. By decentralizing health analytics, edge computing ensures that critical alerts are generated without delays, enabling timely intervention and reducing dependency on centralized infrastructures [114,115].

Cloud services complement edge processing by providing deep learning (DL) capabilities for disease prediction, medical image analysis, and large-scale epidemiological modeling. The hybrid approach enhances diagnostic accuracy while supporting personalized treatment plans based on historical patient data. FL further strengthens privacy by training AI models across distributed edge nodes, preventing sensitive medical data from being exposed to centralized repositories [116–118].

Emergency response systems leverage edge–cloud architectures to optimize medical resource allocation and dynamic dispatch of ambulances and personnel. AI-driven triage mechanisms assist in prioritizing emergency cases by analyzing real-time patient conditions, reducing response times, and improving survival rates. These innovations collectively enable intelligent, real-time, and scalable healthcare services, addressing the growing challenges of modern medical infrastructures [119–121].

4.3. Industrial Automation and Smart Manufacturing

The integration of edge–cloud computing in industrial automation enhances predictive maintenance, robotic coordination, and process optimization, significantly improving production efficiency. Edge nodes facilitate localized decision-making by continuously monitoring machine performance, detecting anomalies, and initiating preventive measures to reduce downtime. AI-enhanced fault detection systems ensure that deviations in operational parameters trigger immediate corrective actions, minimizing financial losses [122,123].

Real-time quality control benefits from edge-based vision systems, which identify product defects through high-speed image analysis and AI-assisted pattern recognition. Cloud integration enhances process optimization by aggregating quality metrics across multiple production lines, refining models for defect prediction and performance improvement. The ability to balance real-time processing at the edge with comprehensive cloud analytics ensures optimal manufacturing workflows [124–126].

Collaborative robotics, or cobots, rely on edge intelligence for distributed control, ensuring synchronized operations in automated assembly lines. Real-time data exchange between robotic agents enables adaptive task execution and improves efficiency in dynamic production environments. The integration of industrial AI, cloud-driven analytics, and distributed edge computing creates highly flexible and autonomous manufacturing ecosystems [127,128].

4.4. Smart Cities and IoT-Based Urban Management

The increasing deployment of IoT devices in urban environments has transformed smart-city management, allowing real-time monitoring of environmental parameters, traffic regulation, and automated public services. Edge computing enables localized processing of urban data streams, ensuring faster decision-making in applications such as smart grids, intelligent waste management, and real-time infrastructure monitoring [129,130].

Cloud computing enhances large-scale urban planning by aggregating historical and real-time data, providing predictive analytics for energy demand, traffic optimization, and air quality management. The hybrid edge–cloud framework ensures that time-sensitive decisions, such as adjusting traffic lights during peak hours or detecting environmental hazards, are handled at the edge while comprehensive analysis and governance remain cloud-centric [131,132].

Energy grid optimization benefits from edge intelligence, where real-time monitoring of consumption patterns enables adaptive load balancing. AI-driven demand–response mechanisms adjust power distribution based on usage trends, improving grid stability and sustainability. Environmental monitoring leverages IoT sensors deployed across urban regions, providing real-time data on air pollution, noise levels, and climate conditions. These insights drive proactive urban policymaking, ensuring sustainable and resilient smart-city ecosystems [133–135].

4.5. Smart Energy and Power Systems

The modernization of energy infrastructures relies on edge–cloud architectures for intelligent grid management, decentralized energy trading, and predictive maintenance. Edge computing enables real-time load balancing by continuously monitoring power consumption, detecting fluctuations, and dynamically adjusting distribution. This decentralized approach enhances grid resilience, preventing failures and optimizing resource utilization [136–138].

Cloud analytics are crucial for predicting energy demand, optimizing renewable energy integration, and improving fault tolerance in power networks. AI-based predictive models refine energy consumption strategies by analyzing historical and real-time grid data, supporting efficient energy allocation [139,140].

The emergence of peer-to-peer energy trading platforms powered by blockchain and edge intelligence enables consumers to exchange surplus electricity securely. Smart contracts enforce automated transactions, reducing reliance on centralized power distribution authorities while fostering decentralized energy markets [141,142].

4.6. Edge-Assisted Augmented and Virtual Reality

The adoption of AR/VR applications demands ultra-low-latency processing and high computational efficiency, making edge–cloud integration essential for immersive experiences. Edge computing accelerates real-time rendering by offloading processing from end-user devices, ensuring seamless motion tracking, adaptive scene generation, and AI-driven interaction modeling [143–145].

Cloud services complement edge computing by handling computationally intensive physics simulations, DL-based content generation, and large-scale data synchronization. This balance ensures that AR/VR experiences remain fluid and responsive, avoiding delays that could degrade user immersion [146,147].

Edge-assisted AI prediction enhances AR/VR experiences by anticipating user movements, reducing perceived latency, and improving interactivity. Optimized network bandwidth allocation further ensures smooth multi-user collaboration in virtual environments, preventing congestion and maintaining synchronization in real-time simulations [148–150].

4.7. Disaster Management and Emergency Response

Edge–cloud computing significantly enhances disaster response by providing real-time situational awareness, predictive analytics, and rapid resource deployment. Edge nodes facilitate immediate hazard detection by processing sensor data from surveillance cameras, unmanned aerial vehicles (UAVs), and environmental sensors. Instantaneous risk assessment enables authorities to make informed decisions and deploy emergency resources efficiently [151–153].

Cloud services assist in large-scale coordination by aggregating multi-source data, refining disaster prediction models, and optimizing evacuation strategies. AI-driven emergency response systems prioritize rescue operations based on real-time impact assessments, ensuring that relief efforts are allocated to the most affected regions [154,155].

UAVs equipped with edge processors play a vital role in disaster monitoring. They assist in search and rescue missions through AI-enhanced object recognition. These advancements ensure faster response times, improved victim detection, and efficient resource utilization in crisis scenarios [156,157].

4.8. Cybersecurity and Threat Detection

As edge–cloud infrastructures expand, ensuring robust cybersecurity is critical to mitigate risks associated with data breaches, unauthorized access, and cyberattacks. AI-driven anomaly detection enables real-time threat identification at the edge, preventing security breaches before they escalate [158–160].

FL enhances cybersecurity by enabling real-time threat intelligence sharing without exposing raw data, improving collaborative defense mechanisms across distributed networks. Blockchain-based security frameworks introduce decentralized trust models, ensuring tamper-proof authentication and secure data exchanges [161–163].

The integration of zero-trust architectures enforces continuous authentication and access verification, reducing vulnerabilities in edge–cloud ecosystems. These advancements collectively strengthen cybersecurity resilience, ensuring the integrity, confidentiality, and availability of edge–cloud services [164,165].

Table 4 provides a summary of various application domains in edge–cloud computing, categorizing them based on their primary objectives, computational challenges, key performance metrics, edge–cloud dependencies, and critical constraints. The table encapsulates how different sectors leverage edge–cloud frameworks to enhance efficiency, minimize latency, and optimize resource management.

Table 4. Summary of application domains in edge–cloud computing.

Application Domain	Primary Objective	Computational Challenges	Key Performance Metrics	Edge-Cloud Dependency	Critical Constraints
Smart transportation [106–113]	Real-time traffic management, autonomous mobility, and safety enhancement.	High-speed vehicular data processing, low-latency V2X communication.	Route optimization time, accident avoidance rate, latency minimization.	Edge for real-time vehicle control and cloud for long-term traffic analytics.	Stringent safety requirements, dynamic network conditions.
Smart healthcare [114–121]	Remote patient monitoring, medical diagnostics, and emergency response.	AI-based anomaly detection, real-time alerting, and privacy preservation.	Detection accuracy, emergency response time, and medical resource availability.	Edge for immediate health data processing, cloud for historical medical trends.	Regulatory compliance, data security, reliability of edge health models.
Industrial automation [122–128]	Predictive maintenance, robotic automation, and process optimization.	Machine status monitoring, robotic coordination, AI-driven analytics.	Fault prediction accuracy, production efficiency, robotic synchronization.	Edge for real-time factory automation, cloud for predictive maintenance.	Synchronization issues in automated systems, cybersecurity risks.
Smart cities [129–135]	Environmental monitoring, traffic regulation, and automated governance.	Distributed sensor fusion, IoT-based analytics, energy optimization.	Data processing efficiency, service availability, energy consumption control.	Edge for localized city services, cloud for policy planning and large-scale governance.	Scalability of IoT networks, energy efficiency, infrastructure costs.
Smart energy [136–142]	Energy grid optimization, decentralized trading, and renewable integration.	Smart contract-based trading, load balancing, fault-tolerant forecasting.	Grid stability, fault tolerance, power efficiency.	Edge for dynamic demand balancing, cloud for predictive analytics.	Renewable energy fluctuations, cybersecurity in decentralized trading.
AR/VR [143–150]	Immersive real-time experiences, interactive collaboration.	Low-latency rendering, AI-assisted prediction, network congestion management.	Frame rate, response delay, and quality of service (QoS) in interactive sessions.	Edge for real-time frame processing and cloud for complex graphics rendering.	Network latency, power constraints of mobile devices, user experience consistency.
Disaster management [151–157]	Early warning, emergency coordination, and rescue optimization.	AI-based threat detection, UAV-assisted search, large-scale event aggregation.	Mission response time, victim detection rate, disaster resilience.	Edge for UAV-based reconnaissance, cloud for large-scale coordination.	Real-time data reliability and communication stability in crisis environments.
Cybersecurity [158–165]	Real-time threat detection, secure authentication, and data integrity.	FL for intrusion detection and blockchain-based identity verification.	Detection rate, false alarm reduction, access trustworthiness.	Edge for decentralized threat detection, cloud for federated cybersecurity intelligence.	Computational cost of security measures, attack resilience, real-time response speed.

5. Challenges, Open Issues, and Future Research Directions

The rapid evolution of edge–cloud computing in smart cities introduces complex challenges spanning system architecture, resource allocation, network optimization, and security. As these paradigms continue to scale, the growing heterogeneity of devices, dynamic workload demands, and the need for real-time processing create significant constraints. Efficient coordination across architectural layers is essential to maintaining service reliability and performance. Additionally, the increasing frequency of inter-tier interactions amplifies issues such as latency, resource provisioning, and energy consumption. This section identifies and examines critical open research areas, including network efficiency, task offloading, data caching, and energy-aware resource management, providing insights into emerging solutions necessary for the seamless deployment of edge–cloud infrastructures.

5.1. Architectural Complexity and System Integration

The integration of edge and cloud computing in smart cities presents significant architectural challenges due to the heterogeneous nature of computing, networking, and storage resources across different tiers. The multi-tier architecture involves frequent inter-layer interactions, requiring efficient coordination to maintain performance, security, and reliability. The complexity arises from diverse hardware and software environments, interoperability constraints, and the need for dynamic adaptation to fluctuating workloads and network conditions [166,167].

One of the primary challenges is system-wide interoperability, as edge devices, edge servers, and cloud platforms operate on different frameworks, communication protocols, and computing models. The lack of standardized architectures complicates seamless integration, making cross-platform orchestration a critical issue. Existing solutions, such as containerized microservices and service mesh architectures, attempt to address this by providing a unified framework for managing distributed workloads. However, ensuring consistency in service execution, API compatibility, and latency-aware decision-making remains an open challenge [168–170].

Moreover, another significant concern is real-time orchestration across tiers. Edge nodes handle time-sensitive tasks closer to end-users, while cloud servers provide long-term data storage and large-scale computation. The challenge lies in determining the optimal placement of services dynamically to minimize network congestion and maximize responsiveness. While AI-driven orchestration frameworks have been explored, their scalability and adaptability to highly dynamic environments require further investigation [171,172].

Future research should focus on developing standardized orchestration frameworks to ensure seamless system integration across heterogeneous edge–cloud environments. AI-driven workload balancing can dynamically optimize resource allocation and cross-layer interactions, reducing system complexity. Additionally, cross-layer optimization strategies must be explored to enhance adaptability in real-time applications while ensuring efficient communication across software and hardware layers [173–175].

5.2. Resource Allocation in Edge-Cloud Environments

Efficient resource allocation in edge–cloud architectures is critical due to heterogeneous computational capacities, dynamic service demands, and network constraints. Unlike traditional cloud-based infrastructures, where centralized schedulers optimize resource distribution, multi-tier edge–cloud environments require decentralized and adaptive allocation mechanisms that can respond to real-time variations in workload and network conditions [176,177].

One of the main challenges is coordinating resource allocation across tiers. Since edge nodes have limited processing power and storage, prioritizing which tasks should be processed locally and which should be offloaded is non-trivial. Current strategies rely on heuristic-based scheduling, but RL and federated optimization are emerging as promising approaches. Additionally, the trade-off between latency and computational load must be optimized, as overloading edge nodes can lead to increased response times and energy consumption [178–180].

Another key issue is interference in shared-resource environments. When multiple applications compete for edge resources, contention leads to degraded performance. Multi-tenant resource isolation, dynamic slicing, and QoS-aware workload allocation are necessary to ensure fair resource distribution. Moreover, in mobility-driven smart-city applications, handover-aware resource allocation is required to prevent service disruptions during user transitions between edge nodes [181,182].

The unpredictability of workloads in edge–cloud environments necessitates RL-based resource management strategies. Future research should focus on predictive workload balancing and decentralized task scheduling, which leverage AI models to dynamically adjust resources based on demand. FL approaches can further enhance allocation by enabling real-time adaptation while preserving privacy [183–186].

5.3. Task Offloading Strategies

Task offloading in edge–cloud environments must be dynamically optimized to balance latency, energy consumption, and computational efficiency. A major challenge is determining the optimal execution location for each task, as edge devices, edge servers, and cloud platforms have distinct processing capabilities. The decision-making process must account for fluctuating network bandwidth, computation delays, and real-time user mobility [187,188].

Existing approaches to task offloading rely on static thresholds or heuristic models, which fail to adapt to dynamic workloads. More recently, Deep RL (DRL)-based adaptive offloading has been proposed, where an AI model continuously learns optimal offloading policies by analyzing network conditions and device capabilities. However, training efficiency, generalization to unseen network states, and scalability to large-scale deployments remain open issues [189–191].

Another challenge is joint optimization of task partitioning and offloading. Many tasks in smart-city applications are computationally intensive and require partial execution at different layers. Traditional binary offloading (executing a task entirely at one location) is often suboptimal, leading to high transmission delays. Partition-aware offloading models, which split tasks dynamically across cloud and edge layers, need further exploration [192,193].

AI-based offloading policies should be adaptive and context-aware, considering network variability, device constraints, and service deadlines. Future studies should explore hybrid offloading models, combining cloud-based and edge-based decision-making mechanisms. Dynamic learning techniques can be employed to adjust offloading decisions in real-time based on evolving network conditions [194–196].

5.4. Data Caching and Content Distribution

Data caching is essential for reducing redundant transmissions, minimizing latency, and improving system throughput in edge–cloud environments. However, efficient cache placement and eviction strategies remain a major challenge due to the dynamic nature of smart-city applications [197,198].

One key issue is predicting data access patterns. Traditional Least Recently Used (LRU) and Least Frequently Used (LFU) caching policies do not account for real-time variations in data demand. Recent studies propose AI-driven predictive caching, where DL models forecast future requests based on historical data. However, challenges such as scalability, model retraining overhead, and adaptability to unseen traffic patterns remain unsolved [199–201].

Another challenge is cooperative caching in multi-tier architectures. With frequent interactions between cloud and edge nodes, data redundancy across tiers must be minimized while ensuring timely content availability. Hierarchical cache synchronization mechanisms, where edge nodes dynamically adjust caching decisions based on cloud-side intelligence, have shown potential but require further optimization [202,203].

Additionally, cache consistency in highly dynamic environments poses a critical problem. When data are updated at the cloud or an edge server, outdated cached copies can cause stale content delivery. Existing solutions rely on synchronous updates, which introduce delays, or asynchronous replication, which risks inconsistencies. Hybrid cache coherence mechanisms, leveraging edge-consensus algorithms and real-time synchronization techniques, need to be explored to ensure accuracy without compromising efficiency [204,205].

Advancements in AI-driven cache management will improve content placement strategies, ensuring minimal redundancy in distributed storage. Future research should focus on collaborative caching schemes, where multiple edge nodes cooperatively manage storage to optimize retrieval speed and reduce bandwidth consumption. Hierarchical caching frameworks could further enhance data accessibility across multi-tier edge–cloud environments [206–208].

5.5. Network Scalability and Latency Optimization

Edge–cloud architectures rely on high-speed communication networks to facilitate real-time data exchange and deliver low-latency service. However, the scalability of these networks remains a fundamental challenge as the number of connected devices and data-intensive applications continues to rise. The proliferation of latency-sensitive services such as autonomous driving, industrial automation, and remote healthcare necessitates network architectures that can efficiently handle massive data volumes while maintaining ultra-reliable low-latency communications. Traditional cloud-centric networking models struggle to meet these stringent requirements, prompting the need for innovative approaches that leverage edge computing for localized data processing and hierarchical network management [209–212].

The dynamic nature of mobile edge computing environments further complicates latency optimization, as user mobility patterns, network congestion, and fluctuating bandwidth availability introduce unpredictable delays. Conventional routing mechanisms are often inadequate for handling time-sensitive data streams, necessitating adaptive transmission protocols that prioritize critical information while mitigating network bottlenecks. The emergence of 5G and beyond-5G technologies presents promising solutions for enhancing network scalability, yet challenges related to spectrum allocation, interference mitigation, and MEC integration persist. These unresolved issues highlight the importance of developing advanced network architectures capable of dynamically provisioning resources, optimizing data paths, and ensuring seamless connectivity in highly distributed edge–cloud ecosystems [213–217].

The increasing demand for low-latency applications requires 5G and beyond technologies to be integrated with edge–cloud architectures. AI-driven adaptive routing protocols should be developed to mitigate congestion dynamically. Furthermore, MEC can be leveraged to bring computation closer to users, reducing latency and improving scalability [218–220].

5.6. Security, Privacy, and Trust Management

The distributed nature of edge–cloud infrastructures introduces substantial security risks, as decentralized computing resources are exposed to a wide range of cyberthreats. Unlike traditional cloud environments that operate within tightly controlled data centers, edge computing infrastructures are inherently more vulnerable to attacks such as data breaches, man-in-the-middle interceptions, and distributed denial-of-service (DDoS) assaults. The integration of diverse computing nodes across public, private, and hybrid domains complicates the enforcement of uniform security policies and access control mechanisms [221–223].

Data privacy concerns further exacerbate the security landscape, particularly in applications involving sensitive information such as healthcare records, financial transactions, and industrial control systems. The necessity to process user-generated data at the edge raises critical questions regarding data ownership, confidentiality preservation, and compliance with regulatory frameworks such as the General Data Protection Regulation (GDPR). Existing cryptographic techniques often introduce computational overhead that may be impractical for resource-constrained edge devices, necessitating lightweight encryption schemes and privacy-preserving FL models [224–226].

The establishment of trust within heterogeneous edge–cloud environments remains an open challenge, as devices from different vendors, network operators, and service providers must interact within shared computational spaces. Blockchain and distributed ledger technologies have emerged as potential solutions for enhancing trust and transparency in edge–cloud transactions, yet scalability issues and consensus latency hinder their widespread adoption. Addressing these security and trust challenges requires a holistic approach that combines intrusion detection systems, access control frameworks, and secure multi-party computation techniques to fortify edge–cloud infrastructures against evolving cyberthreats [227–230].

To address growing security threats, future research should focus on blockchain-based security models, which provide decentralized trust mechanisms with verifiable immutability. Additionally, FL can be used for privacy-preserving analytics, enabling AI models to train on distributed data without exposing sensitive information. Lightweight encryption techniques should be optimized to enhance security with minimal computational overhead [231–233].

5.7. Resource Management

Efficient resource management in multi-tier edge–cloud architectures is crucial to maintaining low latency, high availability, and optimal service performance in smart-city applications. Unlike traditional cloud-centric models, where resources are centrally managed, edge–cloud environments require decentralized, real-time coordination of computing, networking, and storage resources. The complexity of resource management arises from dynamic workloads, fluctuating network conditions, heterogeneous edge nodes, and mobility-induced service migrations [234,235].

A key challenge is multi-tier resource orchestration, where computational tasks must be dynamically distributed among cloud servers, edge nodes, and end devices based on real-time constraints such as latency, energy efficiency, and network congestion. Existing scheduling mechanisms, such as heuristic-based static allocation and first-come–first-served (FCFS) models, often fail under highly dynamic conditions. RL-based schedulers have emerged as promising solutions, enabling real-time decision-making based on changing system states. However, scalability issues, convergence speed, and high training overhead limit their practical deployment [236–238].

Another challenge is cross-layer resource optimization, where computing, communication, and storage resources must be jointly managed to ensure end-to-end service continuity. Traditional solutions treat these layers independently, leading to suboptimal performance. Recent research focuses on DL-driven cross-layer schedulers, which optimize central processing unit (CPU) cycles, bandwidth allocation, and memory utilization simultaneously. However, the real-time execution of these models is computationally expensive, necessitating lightweight approximation methods that maintain high accuracy without excessive processing overhead [239,240].

Mobility-aware resource allocation is particularly challenging in smart-city environments where users frequently switch between edge nodes. Traditional fixed allocation approaches struggle to maintain seamless service continuity due to handover-induced delays. Emerging solutions, such as proactive migration strategies based on graph neural networks (GNNs), attempt to predict user movement and reallocate resources accordingly. However, achieving high prediction accuracy without excessive computation remains an open issue [241–243].

Interference management in multi-tenant edge environments is another critical concern. As multiple services compete for limited resources, contention can degrade performance and violate QoS agreements. Current approaches use resource slicing and priority-based queuing mechanisms, but fine-grained control remains a challenge [244,245].

Future advancements should incorporate AI-powered workload scheduling to ensure efficient utilization of computing resources across dynamic edge–cloud networks. Decentralized resource orchestration models should also be explored to minimize bottlenecks and improve responsiveness in high-load environments [246–248].

5.8. Energy Efficiency

Energy efficiency is a critical concern in edge–cloud computing due to the resource constraints of edge nodes and the high power consumption of cloud data centers. Unlike cloud environments, where centralized power management can be optimized at scale, edge nodes operate in distributed, often energy-limited environments, making fine-grained energy control essential. Smart-city applications, including real-time surveillance, autonomous transportation, and industrial automation, require continuous data processing at the edge, leading to high energy demands that must be optimized without sacrificing performance [249–251].

One of the main challenges is energy-aware task scheduling and workload balancing across cloud, edge, and end devices. DVFS has been widely adopted to adjust processing power based on workload demands, but its effectiveness is limited in latency-sensitive applications. A more adaptive solution involves RL-based power management, where an AI model continuously learns optimal CPU/GPU scaling strategies based on real-time workload variations. However, training such models remains computationally expensive, and their deployment at the edge requires lightweight inference models to minimize processing overhead [252–254].

Another issue is energy-efficient communication between edge nodes and cloud servers. Frequent data transmissions over wireless networks consume significant power, particularly in mobile environments. Emerging solutions leverage adaptive edge caching and data compression techniques to reduce redundant transmissions, thereby lowering energy consumption. Additionally, energy-aware network slicing can optimize resource allocation at the network level, ensuring that only the required computing resources are activated while deactivating idle components [255–257].

Furthermore, heterogeneous energy consumption in multi-tier architectures presents optimization challenges. While cloud servers can leverage liquid cooling and advanced thermal management systems, edge devices must rely on low-power hardware accelerators such as ARM-based processors, Field Programmable Gate Arrays (FPGAs), and neuromorphic chips to achieve energy efficiency. However, the integration of specialized hardware into existing edge frameworks remains a challenge due to compatibility issues and software–hardware co-design constraints [258,259].

Dynamic energy-aware workload scheduling should be implemented to optimize power consumption without compromising performance. Future research should explore low-power computing techniques, including neuromorphic processing and green energy integration, to enhance sustainability in edge–cloud environments. Additionally, predictive task migration mechanisms can be employed to minimize energy-intensive operations [260–263].

5.9. Standardization and Interoperability Constraints

The absence of standardized protocols and interoperability frameworks poses a significant barrier to the widespread adoption of edge–cloud computing. The fragmented nature of current implementations results in compatibility issues across different platforms, leading to inefficiencies in system deployment and operational management. The lack of universal communication standards hinders seamless interaction between heterogeneous devices, making it difficult to achieve cohesive and scalable edge–cloud infrastructures [264–266].

The integration of edge computing with emerging technologies such as 6G, blockchain, and AI-driven decision-making further amplifies the need for standardized architectures. Existing frameworks often fail to address the dynamic requirements of real-time edge processing, necessitating flexible and adaptive standards that can accommodate evolving computing paradigms. The challenge lies in defining interoperability guidelines that enable diverse edge–cloud environments to function cohesively while ensuring compliance with regulatory mandates and industry best practices [267–269].

Developing globally accepted edge–cloud standards requires collaboration among industry leaders, academic researchers, and regulatory bodies. The establishment of unified frameworks for workload orchestration, data exchange, and security enforcement will be essential in enabling large-scale deployments while reducing integration overhead. Addressing these standardization constraints will play a pivotal role in shaping the future of edge–cloud computing, ensuring that heterogeneous infrastructures can seamlessly interoperate across diverse application domains [270–272].

Future research should prioritize the development of universal communication protocols that facilitate seamless interoperability across diverse platforms. Cross-industry collaborations will be essential for establishing regulatory frameworks and compliance standards to ensure consistent and scalable edge–cloud deployments [273–275].

In conclusion, Table 5 provides a comparative summary of challenges in edge–cloud computing, outlining key issues, their impact, potential solutions, and future research directions. It highlights areas such as architectural complexity, resource allocation, security, and energy efficiency while suggesting AI-driven optimizations, decentralized models, and emerging technologies (e.g., 6G, blockchain, neuromorphic computing) to enhance scalability, efficiency, and interoperability.

Table 5. Comparative summary of challenges, key issues, impact, potential solutions, and future research directions in edge–cloud computing.

Challenge	Key Issues	Impact on Edge-Cloud	Potential Solutions	Future Directions
Architectural complexity and system integration [166–172]	Heterogeneous hardware and software platforms, inefficient workload distribution, cross-layer dependency management.	Increased complexity in system deployment, suboptimal resource utilization, and reduced adaptability in real-time applications.	Standardized orchestration frameworks, intelligent workload balancing, and cross-layer optimization strategies.	Developing AI-driven self-adaptive orchestration for real-time workload distribution and cross-platform interoperability using GNNs. [173–175]
Resource allocation in edge–cloud environments [176–182]	Dynamic workload distribution, inefficient resource provisioning, unpredictable demand fluctuations.	Service degradation, increased response times, excessive energy consumption in high-load scenarios.	RL-based resource management, predictive workload balancing, decentralized task scheduling.	Hybrid learning models for dynamic resource allocation, integrating FL to enhance distributed decision-making. [183–186]
Task offloading strategies [187–193]	Suboptimal decision-making in offloading strategies, high communication overhead, network variability effects.	Increased latency, excessive energy drain in mobile edge devices, inefficient execution of real-time applications.	AI-based offloading policies, adaptive learning techniques, edge-to-cloud migration frameworks.	Exploring multi-agent RL for intelligent cooperative offloading in dynamic network conditions. [194–196]
Data caching and content distribution [197–205]	Redundant data transmissions, inefficient caching policies, limited storage in edge nodes.	Increased bandwidth consumption, high data retrieval delays, inconsistent caching effectiveness.	AI-driven cache management, collaborative caching schemes, hierarchical caching frameworks.	Using edge-aware predictive caching mechanisms with DL to improve data retrieval efficiency. [206–208]
Network scalability and latency optimization [209–217]	Scalability limitations, high latency in dynamic environments, network congestion, suboptimal routing protocols.	Inability to handle large-scale data processing, reduced QoS for latency-sensitive applications, inconsistent service delivery.	5G and beyond networks, AI-driven adaptive routing, MEC integration.	Leveraging 6G networks and quantum-assisted computing to optimize ultra-low-latency communications. [218–220]
Security, privacy, and trust management [221–230]	Exposure to cyberthreats, data privacy concerns, decentralized trust enforcement, security overhead in edge nodes.	High risk of data breaches, increased computational costs for security enforcement, reduced user trust in distributed systems.	Blockchain-based security models, FL for privacy-preserving analytics, lightweight encryption schemes.	Integrating homomorphic encryption and zero-trust architectures to ensure secure decentralized processing. [231–233]
Resource management [234–245]	Inefficient resource allocation, lack of adaptive scaling mechanisms, poor cross-domain resource sharing.	Suboptimal resource utilization, service bottlenecks, and reduced performance in dynamic environments.	Decentralized resource scheduling, multi-agent resource optimization techniques.	Developing AI-driven intent-based resource allocation frameworks that autonomously adjust to workload shifts. [246–248]
Energy efficiency [249–259]	High energy consumption in constrained environments, inefficient power allocation, unpredictable workload energy demands.	Increased operational costs, sustainability concerns, performance bottlenecks in mobile and IoT-based applications.	AI-powered workload scheduling, dynamic energy scaling techniques, predictive task migration mechanisms.	Exploring neuromorphic computing and energy-aware AI models to minimize power consumption in edge–cloud infrastructures. [260–263]
Standardization and interoperability constraints [264–272]	Lack of unified standards, interoperability issues across different platforms, regulatory compliance challenges.	Fragmentation in edge–cloud deployments, difficulty in achieving seamless integration, increased operational overhead.	Development of universal communication protocols, industry-wide collaboration for standardization, adaptive compliance frameworks.	Creating a globally accepted edge–cloud standardization framework with cross-industry collaboration. [273–275]

6. Conclusions

The findings of this survey reveal that the integration of edge and cloud computing plays a pivotal role in shaping the future of smart cities, enabling real-time analytics, resource-efficient computation, and intelligent decision-making. Through an extensive examination of architectural models, enabling technologies, and diverse application domains, this study demonstrates how edge–cloud infrastructures optimize computational efficiency while minimizing latency and bandwidth overhead. The comparative analysis of various architectural paradigms—ranging from hierarchical multi-tier designs to fully distributed and FL-enhanced frameworks—illustrates that each model presents unique trade-offs concerning scalability, resilience, and energy efficiency. The findings indicate that hybrid approaches, particularly those incorporating digital twins and AI-driven orchestration, offer promising pathways toward adaptive and self-optimizing urban infrastructures.

Moreover, the enabling technologies explored in this survey underscore the significance of advanced networking protocols, AI-based resource management, blockchain security, and FL in augmenting the performance and security of edge–cloud ecosystems. High-speed communication networks, such as 5G and future 6G architectures, provide ultra-low-latency data transmission essential for real-time applications, while federated intelligence facilitates decentralized learning models that enhance privacy preservation. However, challenges related to synchronization, interoperability, and security enforcement remain key obstacles that necessitate further investigation. The survey findings emphasize that future research must focus on developing robust mechanisms for workload balancing, real-time fault tolerance, and energy-efficient computing to ensure sustainable deployment in large-scale urban environments.

Application-specific insights from smart transportation, healthcare, industrial automation, and urban IoT management further reinforce the practical relevance of edge–cloud computing in transforming smart-city services. The analysis of these domains highlights the imperative for dynamic workload migration strategies, real-time AI inferencing, and secure data-sharing mechanisms to accommodate the diverse computational needs of intelligent infrastructures. While edge-assisted architectures successfully reduce latency for time-sensitive applications, cloud-based analytics remain indispensable for large-scale data aggregation and long-term predictive modeling. The findings of this survey strongly suggest that an optimal edge–cloud synergy, supported by AI-driven decision-making and next-generation networking, will be instrumental in achieving sustainable, resilient, and highly adaptive smart-city ecosystems.

Author Contributions: E.D. and M.T. conceived of the idea, designed and performed the experiments, analyzed the results, drafted the initial manuscript, and revised the final manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

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

List of Abbreviations

The following abbreviations are used in this manuscript:

Variable/Parameter	**Definition**
C_m	Cloud node m
P	Processing power
S	Available storage
L	Inherent processing latency
T_C	Total execution latency at the cloud
$T_{D \to C}$	Time required to transmit data from device or edge to the cloud
$T_{C \to D}$	Response time for sending processed results back to the device
E	Set of K edge nodes
E_k	Edge node k
E_{qi}	Edge node i at cluster q

D	Set of N IoT devices in the system
D_n	n-th IoT device
α_i	Execution ratio determining task execution at an edge node
C_τ	Computational demand of a task τ
C_{E_i}	Computational capacity at edge node E_i
C_C	Computational capacity at the cloud
T_{E_i}	Processing delay at edge node E_i
$F(\tau)$	Decision function determining execution location of task τ
Dt_i	Dataset at edge node i for training AI models
$\mathcal{M}_i^t$	Updated model at edge node i in training round t
η	Learning rate for model updates
T_{sys}	Total system latency in a fully distributed edge–cloud network
d_{E_j,E_k}	Distance between two edge nodes (j,k)
B_{E_j,E_k}	Available bandwidth for communication between two edge nodes (j,k)
P_f	Failure probability of a task execution
$P_{D\to E_i}$	Power consumed for data transmission from devices to edge
$P_{E_i}^{active}$	Power consumed when processing at the edge
$P_{E_i}^{idle}$	Power consumed when idle at the edge
$P_{E_i\to C}$	Power consumed for cloud communication

References

1. Sathupadi, K.; Achar, S.; Bhaskaran, S.V.; Faruqui, N.; Abdullah-Al-Wadud, M.; Uddin, J. Edge-cloud synergy for AI-enhanced sensor network data: A real-time predictive maintenance framework. *Sensors* **2024**, *24*, 7918. [CrossRef] [PubMed]
2. Li, C.; Sun, H.; Tang, H.; Luo, Y. Adaptive resource allocation based on the billing granularity in edge-cloud architecture. *Comput. Commun.* **2019**, *145*, 29–42. [CrossRef]
3. Wang, K.; Jin, J.; Yang, Y.; Zhang, T.; Nallanathan, A.; Tellambura, C.; Jabbari, B. Task offloading with multi-tier computing resources in next generation wireless networks. *IEEE J. Sel. Areas Commun.* **2022**, *41*, 306–319. [CrossRef]
4. Huang, Z.; Shen, Y.; Li, J.; Fey, M.; Brecher, C. A survey on AI-driven digital twins in industry 4.0: Smart manufacturing and advanced robotics. *Sensors* **2021**, *21*, 6340. [CrossRef]
5. Truong, N.; Lee, G.M.; Sun, K.; Guitton, F.; Guo, Y. A blockchain-based trust system for decentralised applications: When trustless needs trust. *Future Gener. Comput. Syst.* **2021**, *124*, 68–79. [CrossRef]
6. Badidi, E. Edge AI and blockchain for smart sustainable cities: Promise and potential. *Sustainability* **2022**, *14*, 7609. [CrossRef]
7. Singh, A.; Satapathy, S.C.; Roy, A.; Gutub, A. Ai-based mobile edge computing for iot: Applications, challenges, and future scope. *Arab. J. Sci. Eng.* **2022**, *47*, 9801–9831. [CrossRef]
8. Tong, Z.; Ye, F.; Yan, M.; Liu, H.; Basodi, S. A survey on algorithms for intelligent computing and smart city applications. *Big Data Min. Anal.* **2021**, *4*, 155–172. [CrossRef]
9. Khan, L.U.; Yaqoob, I.; Tran, N.H.; Kazmi, S.A.; Dang, T.N.; Hong, C.S. Edge-computing-enabled smart cities: A comprehensive survey. *IEEE Internet Things J.* **2020**, *7*, 10200–10232. [CrossRef]
10. Liu, Q.; Gu, J.; Yang, J.; Li, Y.; Sha, D.; Xu, M.; Shams, I.; Yu, M.; Yang, C. Cloud, edge, and mobile computing for smart cities. In *Urban Informatics*; Springer: Singapore, 2021; pp. 757–795.
11. Dave, R.; Seliya, N.; Siddiqui, N. The benefits of edge computing in healthcare, smart cities, and IoT. *arXiv* **2021**, arXiv:2112.01250. [CrossRef]
12. Tufail, A.; Namoun, A.; Alrehaili, A.; Ali, A. A survey on 5G enabled multi-access edge computing for smart cities: Issues and future prospects. *Int. J. Comput. Sci. Netw. Secur.* **2021**, *21*, 107–118.
13. Tahirkheli, A.I.; Shiraz, M.; Hayat, B.; Idrees, M.; Sajid, A.; Ullah, R.; Ayub, N.; Kim, K.I. A survey on modern cloud computing security over smart city networks: Threats, vulnerabilities, consequences, countermeasures, and challenges. *Electronics* **2021**, *10*, 1811. [CrossRef]
14. Li, X.; Abdallah, M.; Lou, Y.Y.; Chiang, M.; Kim, K.T.; Bagchi, S. Dynamic DAG-application scheduling for multi-tier edge computing in heterogeneous networks. *arXiv* **2024**, arXiv:2409.10839.
15. Tao, M.; Zuo, J.; Liu, Z.; Castiglione, A.; Palmieri, F. Multi-layer cloud architectural model and ontology-based security service framework for IoT-based smart homes. *Future Gener. Comput. Syst.* **2018**, *78*, 1040–1051. [CrossRef]
16. Yao, J.; Zhang, S.; Yao, Y.; Wang, F.; Ma, J.; Zhang, J.; Chu, Y.; Ji, L.; Jia, K.; Shen, T.; et al. Edge-cloud polarization and collaboration: A comprehensive survey for ai. *IEEE Trans. Knowl. Data Eng.* **2022**, *35*, 6866–6886. [CrossRef]

17. Zheng, G.; Zhang, H.; Li, Y.; Xi, L. 5G network-oriented hierarchical distributed cloud computing system resource optimization scheduling and allocation. *Comput. Commun.* **2020**, *164*, 88–99. [CrossRef]
18. Chen, C.H.; Liu, C.T. A 3.5-tier container-based edge computing architecture. *Comput. Electr. Eng.* **2021**, *93*, 107227. [CrossRef]
19. Shyam, G.K.; Chandrakar, I. Resource allocation in cloud computing using optimization techniques. In *Cloud Computing for Optimization: Foundations, Applications, and Challenges*; Springer: Cham, Switzerland, 2018; pp. 27–50.
20. Sankaranarayanan, S.; Rodrigues, J.J.; Sugumaran, V.; Kozlov, S. Data flow and distributed deep neural network based low latency IoT-edge computation model for big data environment. *Eng. Appl. Artif. Intell.* **2020**, *94*, 103785.
21. Bozorgchenani, A.; Tarchi, D.; Corazza, G.E. Centralized and distributed architectures for energy and delay efficient fog network-based edge computing services. *IEEE Trans. Green Commun. Netw.* **2018**, *3*, 250–263. [CrossRef]
22. Deng, X.; Yin, J.; Guan, P.; Xiong, N.N.; Zhang, L.; Mumtaz, S. Intelligent delay-aware partial computing task offloading for multiuser industrial Internet of Things through edge computing. *IEEE Internet Things J.* **2021**, *10*, 2954–2966. [CrossRef]
23. Wang, B.; Wang, C.; Huang, W.; Song, Y.; Qin, X. A survey and taxonomy on task offloading for edge-cloud computing. *IEEE Access* **2020**, *8*, 186080–186101. [CrossRef]
24. Chakraborty, C.; Mishra, K.; Majhi, S.K.; Bhuyan, H.K. Intelligent Latency-aware tasks prioritization and offloading strategy in Distributed Fog-Cloud of Things. *IEEE Trans. Ind. Inform.* **2022**, *19*, 2099–2106. [CrossRef]
25. Gu, X.; Zhang, G.; Cao, Y. Cooperative mobile edge computing-cloud computing in Internet of vehicle: Architecture and energy-efficient workload allocation. *Trans. Emerg. Telecommun. Technol.* **2021**, *32*, e4095. [CrossRef]
26. Aslanpour, M.S.; Toosi, A.N.; Cheema, M.A.; Gaire, R. Energy-aware resource scheduling for serverless edge computing. In Proceedings of the 22nd IEEE International Symposium on Cluster, Cloud and Internet Computing (CCGrid), Taormina (Messina), Italy, 16–19 May 2022 ; pp. 190–199.
27. Gill, S.S.; Golec, M.; Hu, J.; Xu, M.; Du, J.; Wu, H.; Walia, G.K.; Murugesan, S.S.; Ali, B.; Kumar, M.; et al. Edge AI: A taxonomy, systematic review and future directions. *Clust. Comput.* **2025**, *28*, 18. [CrossRef]
28. Bukhsh, M.; Abdullah, S.; Bajwa, I.S. A decentralized edge computing latency-aware task management method with high availability for IoT applications. *IEEE Access* **2021**, *9*, 138994–139008. [CrossRef]
29. Wang, C.; Gill, C.; Lu, C. Frame: Fault tolerant and real-time messaging for edge computing. In Proceedings of the 2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS), Dallas, TX, USA, 7–10 July 2019; pp. 976–985.
30. Zeng, L.; Ye, S.; Chen, X.; Zhang, X.; Ren, J.; Tang, J.; Yang, Y.; Shen, X.S. Edge Graph Intelligence: Reciprocally Empowering Edge Networks with Graph Intelligence. *IEEE Commun. Surv. Tutor.* **2025**. [CrossRef]
31. Zhang, Y.; Lan, X.; Ren, J.; Cai, L. Efficient computing resource sharing for mobile edge-cloud computing networks. *IEEE/ACM Trans. Netw.* **2020**, *28*, 1227–1240. [CrossRef]
32. Naouri, A.; Wu, H.; Nouri, N.A.; Dhelim, S.; Ning, H. A novel framework for mobile-edge computing by optimizing task offloading. *IEEE Internet Things J.* **2021**, *8*, 13065–13076. [CrossRef]
33. Sun, M.; Quan, S.; Wang, X.; Huang, Z. Latency-aware scheduling for data-oriented service requests in collaborative IoT-edge-cloud networks. *Future Gener. Comput. Syst.* **2025**, *163*, 107538. [CrossRef]
34. Jiang, H.; Dai, X.; Xiao, Z.; Iyengar, A. Joint task offloading and resource allocation for energy-constrained mobile edge computing. *IEEE Trans. Mob. Comput.* **2022**, *22*, 4000–4015. [CrossRef]
35. Yang, S. A joint optimization scheme for task offloading and resource allocation based on edge computing in 5G communication networks. *Comput. Commun.* **2020**, *160*, 759–768. [CrossRef]
36. Chen, Y.; Zhao, F.; Lu, Y.; Chen, X. Dynamic task offloading for mobile edge computing with hybrid energy supply. *Tsinghua Sci. Technol.* **2022**, *28*, 421–432. [CrossRef]
37. Kapsalis, A.; Kasnesis, P.; Venieris, I.S.; Kaklamani, D.I.; Patrikakis, C.Z. A cooperative fog approach for effective workload balancing. *IEEE Cloud Comput.* **2017**, *4*, 36–45. [CrossRef]
38. Ren, J.; Yu, G.; He, Y.; Li, G.Y. Collaborative cloud and edge computing for latency minimization. *IEEE Trans. Veh. Technol.* **2019**, *68*, 5031–5044. [CrossRef]
39. Van Huynh, D.; Nguyen, V.D.; Chatzinotas, S.; Khosravirad, S.R.; Poor, H.V.; Duong, T.Q. Joint communication and computation offloading for ultra-reliable and low-latency with multi-tier computing. *IEEE J. Sel. Areas Commun.* **2022**, *41*, 521–537. [CrossRef]
40. Aral, A.; Brandić, I. Learning spatiotemporal failure dependencies for resilient edge computing services. *IEEE Trans. Parallel Distrib. Syst.* **2020**, *32*, 1578–1590. [CrossRef]
41. Peng, Q.; Wu, C.; Xia, Y.; Ma, Y.; Wang, X.; Jiang, N. DoSRA: A decentralized approach to online edge task scheduling and resource allocation. *IEEE Internet Things J.* **2021**, *9*, 4677–4692. [CrossRef]
42. Li, S.; Huang, J. Energy efficient resource management and task scheduling for IoT services in edge computing paradigm. In Proceedings of the 2017 IEEE International Symposium on Parallel and Distributed Processing with Applications and 2017 IEEE International Conference on Ubiquitous Computing and Communications (ISPA/IUCC), Guangzhou, China, 12–15 December 2017; pp. 846–851.

43. Xu, J.; Xu, Z.; Shi, B. Deep reinforcement learning based resource allocation strategy in cloud-edge computing system. *Front. Bioeng. Biotechnol.* **2022**, *10*, 908056.
44. Habak, K.; Zegura, E.W.; Ammar, M.; Harras, K.A. Workload management for dynamic mobile device clusters in edge femtoclouds. In Proceedings of the Second ACM/IEEE Symposium on Edge Computing, San Jose, CA, USA, 12–14 October 2017; pp. 1–14.
45. Queralta, J.P.; Westerlund, T. Blockchain for mobile edge computing: Consensus mechanisms and scalability. In *Mobile Edge Computing*; Springer: Cham, Switzerland, 2021; pp. 333–357.
46. Li, C.; Yang, H.; Sun, Z.; Yao, Q.; Zhang, J.; Yu, A.; Vasilakos, A.V.; Liu, S.; Li, Y. High-precision cluster federated learning for smart home: An edge-cloud collaboration approach. *IEEE Access* **2023**, *11*, 102157–102168. [CrossRef]
47. Lim, W.Y.B.; Ng, J.S.; Xiong, Z.; Jin, J.; Zhang, Y.; Niyato, D.; Leung, C.; Miao, C. Decentralized edge intelligence: A dynamic resource allocation framework for hierarchical federated learning. *IEEE Trans. Parallel Distrib. Syst.* **2021**, *33*, 536–550. [CrossRef]
48. Liu, Q.; Cheng, L.; Ozcelebi, T.; Murphy, J.; Lukkien, J. Deep reinforcement learning for IoT network dynamic clustering in edge computing. In Proceedings of the 2019 19th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID), Larnaca, Cyprus, 14–17 May 2019; pp. 600–603.
49. Zhou, Y.; Pang, X.; Wang, Z.; Hu, J.; Sun, P.; Ren, K. Towards efficient asynchronous federated learning in heterogeneous edge environments. In Proceedings of the IEEE INFOCOM 2024-IEEE Conference on Computer Communications, Vancouver, BC, Canada, 20–23 May 2024; pp. 2448–2457.
50. Zhang, T.; Liu, C.; Tian, Q.; Cheng, B. Cloud-Edge Collaboration-Based Multi-Cluster System for Space-Ground Integrated Network. *Int. J. Satell. Commun. Netw.* **2025**, *43*, 40–60. [CrossRef]
51. Wu, B.; Zeng, J.; Ge, L.; Su, X.; Tang, Y. Energy-latency aware offloading for hierarchical mobile edge computing. *IEEE Access* **2019**, *7*, 121982–121997. [CrossRef]
52. Li, C.; Bai, J.; Tang, J. Joint optimization of data placement and scheduling for improving user experience in edge computing. *J. Parallel Distrib. Comput.* **2019**, *125*, 93–105. [CrossRef]
53. Liu, W.; Xu, X.; Li, D.; Qi, L.; Dai, F.; Dou, W.; Ni, Q. Privacy preservation for federated learning with robust aggregation in edge computing. *IEEE Internet Things J.* **2022**, *10*, 7343–7355. [CrossRef]
54. Liu, L.; Zhang, J.; Song, S.; Letaief, K.B. Client-edge-cloud hierarchical federated learning. In Proceedings of the ICC 2020–2020 IEEE International Conference on Communications (ICC), Dublin, Ireland, 7–11 June 2020; pp. 1–6.
55. Lin, F.P.C.; Brinton, C.G.; Michelusi, N. Federated learning with communication delay in edge networks. In Proceedings of the GLOBECOM. IEEE Global Communications Conference, Taipei, Taiwan, 7–11 December 2020; pp. 1–6.
56. Javed, A.; Heljanko, K.; Buda, A.; Främling, K. CEFIoT: A fault-tolerant IoT architecture for edge and cloud. In Proceedings of the 2018 IEEE 4th World Forum on Internet of Things (WF-IoT), Singapore, 5–8 February 2018; pp. 813–818.
57. Mughal, F.R.; He, J.; Das, B.; Dharejo, F.A.; Zhu, N.; Khan, S.B.; Alzahrani, S. Adaptive federated learning for resource-constrained IoT devices through edge intelligence and multi-edge clustering. *Sci. Rep.* **2024**, *14*, 28746. [CrossRef]
58. Márquez-Sánchez, S.; Calvo-Gallego, J.; Erbad, A.; Ibrar, M.; Fernandez, J.H.; Houchati, M.; Corchado, J.M. Enhancing building energy management: Adaptive edge computing for optimized efficiency and inhabitant comfort. *Electronics* **2023**, *12*, 4179. [CrossRef]
59. Crespo-Aguado, M.; Lozano, R.; Hernandez-Gobertti, F.; Molner, N.; Gomez-Barquero, D. Flexible Hyper-Distributed IoT–Edge–Cloud Platform for Real-Time Digital Twin Applications on 6G-Intended Testbeds for Logistics and Industry. *Future Internet* **2024**, *16*, 431. [CrossRef]
60. Dai, Y.; Zhang, Y. Adaptive digital twin for vehicular edge computing and networks. *J. Commun. Inf. Networks* **2022**, *7*, 48–59. [CrossRef]
61. Savaglio, C.; Barbuto, V.; Awan, F.M.; Minerva, R.; Crespi, N.; Fortino, G. Opportunistic digital twin: An edge intelligence enabler for smart city. In *ACM Transactions on Sensor Networks*; Association for Computing Machinery: New York, NY, USA, 2023.
62. Tan, A. Predictive Analytics and Simulation for Digital Twin-enabled Decision Support in Smart Cities. *J. Comput. Soc. Dyn.* **2023**, *8*, 52–62.
63. Leng, J.; Zhang, H.; Yan, D.; Liu, Q.; Chen, X.; Zhang, D. Digital twin-driven manufacturing cyber-physical system for parallel controlling of smart workshop. *J. Ambient Intell. Humaniz. Comput.* **2019**, *10*, 1155–1166. [CrossRef]
64. Shin, H.; Kwak, Y. Enhancing digital twin efficiency in indoor environments: Virtual sensor-driven optimization of physical sensor combinations. *Autom. Constr.* **2024**, *161*, 105326. [CrossRef]
65. Zhang, Y.; Hu, J.; Min, G. Digital twin-driven intelligent task offloading for collaborative mobile edge computing. *IEEE J. Sel. Areas Commun.* **2023**, *40*, 3034–3045. [CrossRef]
66. Younis, A.; Qiu, B.; Pompili, D. Latency-aware hybrid edge cloud framework for mobile augmented reality applications. In Proceedings of the 2020 17th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON), Como, Italy, 22–25 June 2020; pp. 1–9.

67. Pan, Y.; Qu, T.; Wu, N.; Khalgui, M.; Huang, G. Digital twin based real-time production logistics synchronization system in a multi-level computing architecture. *J. Manuf. Syst.* **2021**, *58*, 246–260. [CrossRef]
68. Jiang, Y.; Li, M.; Li, M.; Liu, X.; Zhong, R.Y.; Pan, W.; Huang, G.Q. Digital twin-enabled real-time synchronization for planning, scheduling, and execution in precast on-site assembly. *Autom. Constr.* **2022**, *141*, 104397. [CrossRef]
69. Zhang, W.Z.; Elgendy, I.A.; Hammad, M.; Iliyasu, A.M.; Du, X.; Guizani, M.; Abd El-Latif, A.A. Secure and optimized load balancing for multitier IoT and edge-cloud computing systems. *IEEE Internet Things J.* **2020**, *8*, 8119–8132. [CrossRef]
70. Ullah, I.; Lim, H.K.; Seok, Y.J.; Han, Y.H. Optimizing task offloading and resource allocation in edge-cloud networks: A DRL approach. *J. Cloud Comput.* **2023**, *12*, 112. [CrossRef]
71. Feng, H.; Qiao, L.; Lv, Z. Innovative soft computing-enabled cloud optimization for next-generation IoT in digital twins. *Appl. Soft Comput.* **2023**, *136*, 110082. [CrossRef]
72. Bousnina, D.; Guerassimoff, G. Optimal energy management in smart energy systems: A deep reinforcement learning approach and a digital twin case-study. *Smart Energy* **2024**, *16*, 100163. [CrossRef]
73. Qu, Y.; Yu, S.; Gao, L.; Sood, K.; Xiang, Y. Blockchained Dual-Asynchronous Federated Learning Services for Digital Twin Empowered Edge-Cloud Continuum. *IEEE Trans. Serv. Comput.* **2024**, *17*, 836–849. [CrossRef]
74. Alghamdi, W.; Albassam, E. Synchronization Patterns for Digital Twin Systems. *J. Appl. Data Sci.* **2024**, *5*, 1026–1037. [CrossRef]
75. Kim, W.; Kim, S.; Jeong, J.; Kim, H.; Lee, H.; Youn, B.D. Digital twin approach for on-load tap changers using data-driven dynamic model updating and optimization-based operating condition estimation. *Mech. Syst. Signal Process.* **2022**, *181*, 109471. [CrossRef]
76. Zhao, J.; Ma, Y.; Xia, Y.; Dai, M.; Chen, P.; Long, T.; Shao, S.; Li, F.; Li, Y.; Zeng, F. A novel fault-tolerant approach for dynamic redundant path selection service migration in vehicular edge computing. *Appl. Sci.* **2022**, *12*, 9987. [CrossRef]
77. Sodin, D.; Rudež, U.; Mihelin, M.; Smolnikar, M.; Čampa, A. Advanced edge-cloud computing framework for automated pmu-based fault localization in distribution networks. *Appl. Sci.* **2021**, *11*, 3100. [CrossRef]
78. Jiang, K.; Zhou, H.; Chen, X.; Zhang, H. Mobile edge computing for ultra-reliable and low-latency communications. *IEEE Commun. Stand. Mag.* **2021**, *5*, 68–75. [CrossRef]
79. Gilly, K.; Bernad, C.; Roig, P.J.; Alcaraz, S.; Filiposka, S. End-to-end simulation environment for mobile edge computing. *Simul. Model. Pract. Theory* **2022**, *121*, 102657. [CrossRef]
80. Xia, B.; Kong, F.; Zhou, J.; Tang, X.; Gong, H. A delay-tolerant data transmission scheme for internet of vehicles based on software defined cloud-fog networks. *IEEE Access* **2020**, *8*, 65911–65922. [CrossRef]
81. Zhu, Z.; Li, X.; Chu, Z. Three major operating scenarios of 5G: EMBB, mMTC, URLLC. *Intell. Sens. Commun. Internet Everything* **2022**, *1*, 15–76.
82. Choi, Y.; Aziz, M.R.K.; Cho, K.; Choi, D. Latency-optimal network intelligence services in SDN/NFV-based energy Internet cyberinfrastructure. *IEEE Access* **2019**, *8*, 4485–4499.
83. Hutton, W.J.; McKinnon, A.D.; Hadley, M.D. Software-defined networking traffic engineering process for operational technology networks. *J. Inf. Warf.* **2019**, *18*, 167–181.
84. Gouareb, R.; Friderikos, V.; Aghvami, A.H. Virtual network functions routing and placement for edge cloud latency minimization. *IEEE J. Sel. Areas Commun.* **2018**, *36*, 2346–2357. [CrossRef]
85. Schelstraete, S.; Vázquez, M.M. Signal-to-Interference plus Noise Ratio (Sinr)-Aware Spatial Reuse. U.S. Patent 18/506,949, 16 May 2024.
86. Hu, X.; Wang, L.; Wong, K.K.; Tao, M.; Zhang, Y.; Zheng, Z. Edge and central cloud computing: A perfect pairing for high energy efficiency and low-latency. *IEEE Trans. Wirel. Commun.* **2019**, *19*, 1070–1083. [CrossRef]
87. Fang, C.; Meng, X.; Hu, Z.; Xu, F.; Zeng, D.; Dong, M.; Ni, W. AI-driven energy-efficient content task offloading in cloud-edge-end cooperation networks. *IEEE Open J. Comput. Soc.* **2022**, *3*, 162–171. [CrossRef]
88. Sonmez, C.; Tunca, C.; Ozgovde, A.; Ersoy, C. Machine learning-based workload orchestrator for vehicular edge computing. *IEEE Trans. Intell. Transp. Syst.* **2020**, *22*, 2239–2251. [CrossRef]
89. Sada, A.B.; Khelloufi, A.; Naouri, A.; Ning, H.; Dhelim, S. Energy-Aware Selective Inference Task Offloading for Real-Time Edge Computing Applications. *IEEE Access* **2024**, *12*, 72924–72937. [CrossRef]
90. Wang, L.; Xu, Y.; Xu, H.; Chen, M.; Huang, L. Accelerating decentralized federated learning in heterogeneous edge computing. *IEEE Trans. Mob. Comput.* **2022**, *22*, 5001–5016. [CrossRef]
91. Song, J.; Wang, W.; Gadekallu, T.R.; Cao, J.; Liu, Y. Eppda: An efficient privacy-preserving data aggregation federated learning scheme. *IEEE Trans. Netw. Sci. Eng.* **2022**, *10*, 3047–3057. [CrossRef]
92. Zhou, Z.; Wang, Q.; Li, J.; Li, Z. Resource allocation using deep deterministic policy gradient-based federated learning for multi-access edge computing. *J. Grid Comput.* **2024**, *22*, 59. [CrossRef]
93. Wang, H.; Zhang, J. Blockchain based data integrity verification for large-scale IoT data. *IEEE Access* **2019**, *7*, 164996–165006. [CrossRef]
94. Halgamuge, M.N. Estimation of the success probability of a malicious attacker on blockchain-based edge network. *Comput. Netw.* **2022**, *219*, 109402. [CrossRef]

95. Dong, J.; Zheng, F.; Lin, J.; Liu, Z.; Xiao, F.; Fan, G. Ec-ecc: Accelerating elliptic curve cryptography for edge computing on embedded gpu tx2. *ACM Trans. Embed. Comput. Syst. (TECS)* **2022**, *21*, 1–25. [CrossRef]
96. Dong, J.; Zhang, P.; Sun, K.; Xiao, F.; Zheng, F.; Lin, J. EG-Four: An Embedded GPU-Based Efficient ECC Cryptography Accelerator for Edge Computing. *IEEE Trans. Ind. Inform.* **2022**, *19*, 7291–7300. [CrossRef]
97. Alappat, M.R. *Multifactor Authentication Using Zero Trust*; Rochester Institute of Technology: Rochester, NY, USA, 2023.
98. Naoko, K. Distributed System Access Control for Fuzzy Mathematics and Probability Theory. *Distrib. Process. Syst.* **2022**, *3*, 72–81.
99. Liu, Y.; Lan, D.; Pang, Z.; Karlsson, M.; Gong, S. Performance evaluation of containerization in edge-cloud computing stacks for industrial applications: A client perspective. *IEEE Open J. Ind. Electron. Soc.* **2021**, *2*, 153–168. [CrossRef]
100. Pal, S.; Pattnaik, P.K. A Simulation-based Approach to Optimize the Execution Time and Minimization of Average Waiting Time Using Queuing Model in Cloud Computing Environment. *Int. J. Electr. Comput. Eng.* **2016**, *6*, 743–750.
101. Ma, X.; Zhou, A.; Zhang, S.; Li, Q.; Liu, A.X.; Wang, S. Dynamic task scheduling in cloud-assisted mobile edge computing. *IEEE Trans. Mob. Comput.* **2021**, *22*, 2116–2130. [CrossRef]
102. Liu, B.; Foroozannejad, M.H.; Ghiasi, S.; Baas, B.M. Optimizing power of many-core systems by exploiting dynamic voltage, frequency and core scaling. In Proceedings of the 2015 IEEE 58th International Midwest Symposium on Circuits and Systems (MWSCAS), Fort Collins, CO, USA, 2–5 August 2015; pp. 1–4.
103. Zhang, Z.; Zhao, Y.; Li, H.; Lin, C.; Liu, J. DVFO: Learning-Based DVFS for Energy-Efficient Edge-Cloud Collaborative Inference. *IEEE Trans. Mob. Comput.* **2024**, *23*, 9042–9059. [CrossRef]
104. Van Dinh, D.; Yoon, B.N.; Le, H.N.; Nguyen, U.Q.; Phan, K.D.; Pham, L.D. ICT enabling technologies for smart cities. In Proceedings of the 2020 22nd International Conference on Advanced Communication Technology (ICACT), Chuncheon, Korea, 11–14 February 2020; pp. 1180–1192.
105. Arora, S.; Tewari, A. AI-driven resilience: Enhancing critical infrastructure with edge computing. *Int. J. Curr. Eng. Technol.* **2022**, *12*, 151–157.
106. Kishk, A.M.; Badawy, M.; Ali, H.A.; Saleh, A.I. A new traffic congestion prediction strategy (TCPS) based on edge computing. *Clust. Comput.* **2022**, *25*, 49–75. [CrossRef]
107. Liu, C.; Ke, L. Cloud assisted Internet of things intelligent transportation system and the traffic control system in the smart city. *J. Control Decis.* **2023**, *10*, 174–187. [CrossRef]
108. Jung, C.; Lee, D.; Lee, S.; Shim, D.H. V2X-communication-aided autonomous driving: System design and experimental validation. *Sensors* **2020**, *20*, 2903. [CrossRef]
109. Zhou, S.; Sun, J.; Xu, K.; Wang, G. AI-driven data processing and decision optimization in IoT through edge computing and cloud architecture. *J. AI-Powered Med. Innov.* **2024**, *2*, 64–92. [CrossRef]
110. He, Y.; Wu, B.; Dong, Z.; Wan, J.; Shi, W. Towards C-V2X enabled collaborative autonomous driving. *IEEE Trans. Veh. Technol.* **2023**, *72*, 15450–15462. [CrossRef]
111. Munir, A.; Blasch, E.; Kwon, J.; Kong, J.; Aved, A. Artificial intelligence and data fusion at the edge. *IEEE Aerosp. Electron. Syst. Mag.* **2021**, *36*, 62–78. [CrossRef]
112. Qu, Z.; Tang, Y.; Muhammad, G.; Tiwari, P. Privacy protection in intelligent vehicle networking: A novel federated learning algorithm based on information fusion. *Inf. Fusion* **2023**, *98*, 101824. [CrossRef]
113. Park, J.; Samarakoon, S.; Shiri, H.; Abdel-Aziz, M.K.; Nishio, T.; Elgabli, A.; Bennis, M. Extreme ultra-reliable and low-latency communication. *Nat. Electron.* **2022**, *5*, 133–141. [CrossRef]
114. Verma, P.; Fatima, S. Smart healthcare applications and real-time analytics through edge computing. In *Internet of Things Use Cases for the Healthcare Industry*; Springer: Cham, Switzerland, 2020; pp. 241–270.
115. Junaid, S.B.; Imam, A.A.; Abdulkarim, M.; Surakat, Y.A.; Balogun, A.O.; Kumar, G.; Shuaibu, A.N.; Garba, A.; Sahalu, Y.; Mohammed, A.; et al. Recent advances in artificial intelligence and wearable sensors in healthcare delivery. *Appl. Sci.* **2022**, *12*, 10271. [CrossRef]
116. Guan, H.; Yap, P.T.; Bozoki, A.; Liu, M. Federated learning for medical image analysis: A survey. *Pattern Recognit.* **2024**, *151*, 110424. [CrossRef]
117. Haq, S.; Verma, P. Edge-Cloud-Assisted Multivariate Time Series Data-Based VAR and Sequential Encoder–Decoder Framework for Multi-Disease Prediction. *Arab. J. Sci. Eng.* **2025**, 1–21. [CrossRef]
118. Khalid, N.; Qayyum, A.; Bilal, M.; Al-Fuqaha, A.; Qadir, J. Privacy-preserving artificial intelligence in healthcare: Techniques and applications. *Comput. Biol. Med.* **2023**, *158*, 106848. [CrossRef]
119. El-Rashidy, N.; Sedik, A.; Siam, A.I.; Ali, Z.H. An efficient edge/cloud medical system for rapid detection of level of consciousness in emergency medicine based on explainable machine learning models. *Neural Comput. Appl.* **2023**, *35*, 10695–10716. [CrossRef]
120. Abiri, S.; Taheri, L.; Kakhki, B.R.; Shahabian, M.; Ziyaei, M.; Shafa, S.; Hakemi, A. Artificial Intelligence In Emergency Medicine And It Impact On Patient Related Factors. *Int. J. Med Investig.* **2024**, *13*, 1–11.
121. Talaat, F.M. Effective prediction and resource allocation method (EPRAM) in fog computing environment for smart healthcare system. *Multimed. Tools Appl.* **2022**, *81*, 8235–8258. [CrossRef]

122. Kim, H.; Shon, T. Industrial network-based behavioral anomaly detection in AI-enabled smart manufacturing. *J. Supercomput.* **2022**, *78*, 13554–13563. [CrossRef] [PubMed]
123. Hu, L.; Miao, Y.; Wu, G.; Hassan, M.M.; Humar, I. iRobot-Factory: An intelligent robot factory based on cognitive manufacturing and edge computing. *Future Gener. Comput. Syst.* **2019**, *90*, 569–577. [CrossRef]
124. Feng, Y.; Yang, C.; Wang, T.; Zheng, H.; Gao, Y.; Fan, W. Quality control system of automobile bearing production based on edge cloud collaboration. In Proceedings of the 2020 International Conference on Advanced Mechatronic Systems (ICAMechS), Hanoi, Vietnam, 10–13 December 2020; pp. 319–322.
125. Okuyelu, O.; Adaji, O. AI-Driven Real-time Quality Monitoring and Process Optimization for Enhanced Manufacturing Performance. *J. Adv. Math. Comput. Sci.* **2024**, *39*, 81–89. [CrossRef]
126. Cociorva, A.; Onofrei, N.; Vîlcea, A.L. Tool Integrations for Monitoring Solutions and Associated Performance Analysis. In Proceedings of the International Conference on Business Excellence, Bucharest, Romania, 20–22 March 2023; Volume 17, pp. 1929–1943.
127. Sharma, N.; Cupek, R. Real-time control and optimization of internal logistics systems with collaborative robots. *Procedia Comput. Sci.* **2023**, *225*, 248–258. [CrossRef]
128. Singh, K.D.; Singh, P.D. Fog-based Edge AI for Robotics: Cutting-edge Research and Future Directions. *EAI Endorsed Trans. AI Robot.* **2023**, *2*. [CrossRef]
129. Gheisari, M.; Pham, Q.V.; Alazab, M.; Zhang, X.; Fernández-Campusano, C.; Srivastava, G. ECA: An edge computing architecture for privacy-preserving in IoT-based smart city. *IEEE Access* **2019**, *7*, 155779–155786. [CrossRef]
130. Barthélemy, J.; Verstaevel, N.; Forehead, H.; Perez, P. Edge-computing video analytics for real-time traffic monitoring in a smart city. *Sensors* **2019**, *19*, 2048. [CrossRef]
131. Belcastro, L.; Marozzo, F.; Orsino, A.; Talia, D.; Trunfio, P. Edge-cloud continuum solutions for urban mobility prediction and planning. *IEEE Access* **2023**, *11*, 38864–38874. [CrossRef]
132. Zhang, L.; Zhou, Z.; Yi, B.; Wang, J.; Chen, C.M.; Shi, C. Edge-Cloud Framework for Vehicle-Road Cooperative Traffic Signal Control in Augmented Internet of Things. *IEEE Internet Things J.* **2024**, *12*, 5488–5499. [CrossRef]
133. Chen, X.; Wen, H.; Ni, W.; Zhang, S.; Wang, X.; Xu, S.; Pei, Q. Distributed online optimization of edge computing with mixed power supply of renewable energy and smart grid. *IEEE Trans. Commun.* **2021**, *70*, 389–403. [CrossRef]
134. Deshpande, V. Smart Grids Integration with AI-Powered Demand Response. *Res. J. Comput. Syst. Eng.* **2024**, *5*, 45–58.
135. Shah, J.; Mishra, B. IoT enabled environmental monitoring system for smart cities. In Proceedings of the 2016 International Conference on Internet of Things and Applications (IOTA), Pune, India, 22–24 January 2016; pp. 383–388.
136. Alorf, A. Edge-Cloud Computing for Scheduling the Energy Consumption in Smart Grid. *Comput. Syst. Sci. Eng.* **2023**, *46*, 273–286. [CrossRef]
137. Mofreh, E.; Nguyen, P.; Kok, K. Secure and Efficient Peer-to-Peer Energy Trading: A Case Study on Edge Computing Simulation Platform for Decentralized Decision Making. In Proceedings of the 2024 IEEE International Conference on Environment and Electrical Engineering and 2024 IEEE Industrial and Commercial Power Systems Europe (EEEIC/I&CPS Europe), Rome, Italy, 17–20 June 2024; pp. 1–6.
138. Zhang, L.; Shi, Y.; Wang, D. A Real-Time Lightweight Perceptron for Cloud–Edge Collaborative Predictive Maintenance of Online Service Systems. *IEEE Internet Things J.* **2025**. [CrossRef]
139. Khan, S.U.; Khan, N.; Ullah, F.U.M.; Kim, M.J.; Lee, M.Y.; Baik, S.W. Towards intelligent building energy management: AI-based framework for power consumption and generation forecasting. *Energy Build.* **2023**, *279*, 112705. [CrossRef]
140. Wang, K.; Wu, J.; Zheng, X.; Li, J.; Yang, W.; Vasilakos, A.V. Cloud-edge orchestrated power dispatching for smart grid with distributed energy resources. *IEEE Trans. Cloud Comput.* **2022**, *11*, 1194–1203. [CrossRef]
141. Wongthongtham, P.; Marrable, D.; Abu-Salih, B.; Liu, X.; Morrison, G. Blockchain-enabled Peer-to-Peer energy trading. *Comput. Electr. Eng.* **2021**, *94*, 107299. [CrossRef]
142. Zhou, Z.; Wang, B.; Dong, M.; Ota, K. Secure and efficient vehicle-to-grid energy trading in cyber physical systems: Integration of blockchain and edge computing. *IEEE Trans. Syst. Man Cybern. Syst.* **2019**, *50*, 43–57. [CrossRef]
143. Ganesan, V.; Padmini, V.S.A.; Devi, V.A.; Chowdhury, S.; Srivastava, G.; Amesho, K.T. AR/VR Data Prediction and a Slicing Model for 5G Edge Computing. In *Machine Learning for Mobile Communications*; CRC Press: Boca Raton, FL, USA, 2024; pp. 171–184.
144. Hazarika, A.; Rahmati, M. Towards an evolved immersive experience: Exploring 5G-and beyond-enabled ultra-low-latency communications for augmented and virtual reality. *Sensors* **2023**, *23*, 3682. [CrossRef]
145. Chen, X.; Gao, W.; Chu, Y.; Song, Y. Enhancing interaction in virtual-real architectural environments: A comparative analysis of generative AI-driven reality approaches. *Build. Environ.* **2024**, *266*, 112113. [CrossRef]
146. Srivastava, A.; Jawaid, S.; Singh, R.; Gehlot, A.; Akram, S.V.; Priyadarshi, N.; Khan, B. Imperative role of technology intervention and implementation for automation in the construction industry. *Adv. Civ. Eng.* **2022**, *2022*, 6716987. [CrossRef]

147. Liang, Y.; Li, G.; Zhang, G.; Guo, J.; Liu, Q.; Zheng, J.; Wang, T. Latency Reduction in Immersive Systems through Request Scheduling with Digital Twin Networks in Collaborative Edge Computing. In *ACM Transactions on Sensor Networks*; Association for Computing Machinery: New York, NY, USA, 2024.
148. Madduru, P. Artificial Intelligence as a service in distributed multi access edge computing on 5G extracting data using IoT and including AR/VR for real-time reporting. *Inf. Technol. Ind.* **2021**, *9*, 912–931. [CrossRef]
149. Rohit, T.; Athrij, S.; Gopalan, S. Protocol for Dynamic Load Distributed Low Latency Web-Based Augmented Reality and Virtual Reality. In Proceedings of the International Conference on Computational Intelligence in Communications and Business Analytics, Kalyani, India, 27–28 January 2023; pp. 118–129.
150. Yang, S.; Yang, P.; Chen, J.; Ye, Q.; Zhang, N.; Shen, X. Delay-optimized multi-user VR streaming via end-edge collaborative neural frame interpolation. *IEEE Trans. Netw. Sci. Eng.* **2023**, *11*, 284–298. [CrossRef]
151. d'Oro, E.C.; Colombo, S.; Gribaudo, M.; Iacono, M.; Manca, D.; Piazzolla, P. Modeling and evaluating a complex edge computing based systems: An emergency management support system case study. *Internet Things* **2019**, *6*, 100054. [CrossRef]
152. Khan, A.; Gupta, S.; Gupta, S.K. Emerging UAV technology for disaster detection, mitigation, response, and preparedness. *J. Field Robot.* **2022**, *39*, 905–955. [CrossRef]
153. Hasanuzzaman, M.; Hossain, S.; Shil, S.K. Enhancing disaster management through AI-driven predictive analytics: Improving preparedness and response. *Int. J. Adv. Eng. Technol. Innov.* **2023**, *1*, 533–562.
154. Zou, Q.; Li, G.; Yu, W. Cloud computing based on computational characteristics for disaster monitoring. *Appl. Sci.* **2020**, *10*, 6676. [CrossRef]
155. Hao, H.; Wang, Y. Assessing disaster impact in real time: Data-driven system integrating humans, hazards, and the built environment. *J. Comput. Civ. Eng.* **2021**, *35*, 04021010. [CrossRef]
156. Alsamhi, S.H.; Shvetsov, A.V.; Kumar, S.; Shvetsova, S.V.; Alhartomi, M.A.; Hawbani, A.; Rajput, N.S.; Srivastava, S.; Saif, A.; Nyangaresi, V.O. UAV computing-assisted search and rescue mission framework for disaster and harsh environment mitigation. *Drones* **2022**, *6*, 154. [CrossRef]
157. Jain, N.; Gambhir, A.; Pandey, M. Unmanned Aerial Networks—UAVs and AI. In *Recent Trends in Artificial Intelligence Towards a Smart World: Applications in Industries and Sectors*; Springer: Berlin/Heidelberg, Germany, **2024**; pp. 321–351.
158. Segar, M.; Zolkipli, M.F. A Study On AI-Driven Solutions for Cloud Security Platform. *INTI J.* **2024**, *2024*. [CrossRef]
159. Sabella, D.; Maloor, K.; Smith, N.; Vanderveen, M.; Kourtis, A. Edge Computing Cybersecurity standards: Protecting infrastructure and applications. *IEEE Access* **2024**, *12*, 185328–185335. [CrossRef]
160. Chen, Z.; Wei, S.; Yu, W.; Nguyen, J.H.; Hatcher, W.G. A cloud/edge computing streaming system for network traffic monitoring and threat detection. *Int. J. Secur. Netw.* **2018**, *13*, 169–186.
161. Ghimire, B.; Rawat, D.B. Recent advances on federated learning for cybersecurity and cybersecurity for federated learning for internet of things. *IEEE Internet Things J.* **2022**, *9*, 8229–8249. [CrossRef]
162. Kouicem, D.E.; Imine, Y.; Bouabdallah, A.; Lakhlef, H. Decentralized blockchain-based trust management protocol for the Internet of Things. *IEEE Trans. Dependable Secur. Comput.* **2020**, *19*, 1292–1306. [CrossRef]
163. Lu, Y.; Huang, X.; Zhang, K.; Maharjan, S.; Zhang, Y. Blockchain empowered asynchronous federated learning for secure data sharing in internet of vehicles. *IEEE Trans. Veh. Technol.* **2020**, *69*, 4298–4311. [CrossRef]
164. Almuseelem, W. Continuous and Mutual Lightweight Authentication for Zero-Trust Architecture-Based Security Framework in Cloud-Edge Computing-Based Healthcare 4.0. *J. Theor. Appl. Inf. Technol.* **2024**, *102*, 66–83.
165. Wang, J.; Ni, M.; Wu, F.; Liu, S.; Qin, J.; Zhu, R. Electromagnetic radiation based continuous authentication in edge computing enabled internet of things. *J. Syst. Archit.* **2019**, *96*, 53–61. [CrossRef]
166. Li, J.; Gu, C.; Xiang, Y.; Li, F. Edge-cloud computing systems for smart grid: State-of-the-art, architecture, and applications. *J. Mod. Power Syst. Clean Energy* **2022**, *10*, 805–817. [CrossRef]
167. Kennedy, J.; Sharma, V.; Varghese, B.; Reaño, C. Multi-tier GPU virtualization for deep learning in cloud-edge systems. *IEEE Trans. Parallel Distrib. Syst.* **2023**, *34*, 2107–2123. [CrossRef]
168. Belcastro, L.; Marozzo, F.; Orsino, A.; Presta, A.; Vinci, A. Developing Cross-Platform and Fast-Responsive Applications on the Edge-Cloud Continuum. In Proceedings of the 2024 15th IFIP Wireless and Mobile Networking Conference (WMNC), Venice, Italy, 11–12 November 2024; pp. 88–93.
169. Ganguli, M.; Ranganath, S.; Ravisundar, S.; Layek, A.; Ilangovan, D.; Verplanke, E. Challenges and opportunities in performance benchmarking of service mesh for the edge. In Proceedings of the 2021 IEEE International Conference on Edge Computing (EDGE), Chicago, IL, USA, 5–10 September 2021; pp. 78–85.
170. Arzo, S.T.; Scotece, D.; Bassoli, R.; Devetsikiotis, M.; Foschini, L.; Fitzek, F.H. Softwarized and containerized microservices-based network management analysis with MSN. *Comput. Netw.* **2024**, *254*, 110750. [CrossRef]
171. Ramamoorthi, V. Real-Time Adaptive Orchestration of AI Microservices in Dynamic Edge Computing. *J. Adv. Comput. Syst.* **2023**, *3*, 1–9. [CrossRef]

172. Ahat, B.; Baktır, A.C.; Aras, N.; Altınel, İ.K.; Özgövde, A.; Ersoy, C. Optimal server and service deployment for multi-tier edge cloud computing. *Comput. Netw.* **2021**, *199*, 108393. [CrossRef]
173. Pendyala, S.K. Edge-Cloud Continuum for Ai-Driven remote Patient Monitoring: A Scalable Framework. *J. Data Sci. Inf. Technol.* **2025**, *2*, 66–74.
174. Uddin, R.S.; Manifa, N.Z.; Chakma, L.; Islam, M.M. Cross-Layer Architecture for Energy Optimization of Edge Computing. In Proceedings of the International Conference on Machine Intelligence and Emerging Technologies, Noakhali, Bangladesh, 23–25 September 2022; pp. 687–701.
175. Huang, X.; Yu, R.; Ye, D.; Shu, L.; Xie, S. Efficient workload allocation and user-centric utility maximization for task scheduling in collaborative vehicular edge computing. *IEEE Trans. Veh. Technol.* **2021**, *70*, 3773–3787. [CrossRef]
176. Fan, Y.; Wang, L.; Wu, W.; Du, D. Cloud/edge computing resource allocation and pricing for mobile blockchain: An iterative greedy and search approach. *IEEE Trans. Comput. Soc. Syst.* **2021**, *8*, 451–463. [CrossRef]
177. Lai, Y.C.; Sudyana, D.; Lin, Y.D.; Verkerken, M.; D'hooge, L.; Wauters, T.; Volckaert, B.; De Turck, F. Task assignment and capacity allocation for ML-based intrusion detection as a service in a multi-tier architecture. *IEEE Trans. Netw. Serv. Manag.* **2022**, *20*, 672–683. [CrossRef]
178. Cui, T.; Yang, R.; Fang, C.; Yu, S. Deep reinforcement learning-based resource allocation for content distribution in IoT-edge-cloud computing environments. *Symmetry* **2023**, *15*, 217. [CrossRef]
179. Lin, H.; Xiao, B.; Zhou, X.; Zhang, Y.; Liu, X. A Multi-Tier Offloading Optimization Strategy for Consumer Electronics in Vehicular Edge Computing. *IEEE Trans. Consum. Electron.* **2025**. [CrossRef]
180. Hu, S.; Li, G.; Shi, W. Lars: A latency-aware and real-time scheduling framework for edge-enabled internet of vehicles. *IEEE Trans. Serv. Comput.* **2021**, *16*, 398–411. [CrossRef]
181. Yarkina, N.; Correia, L.M.; Moltchanov, D.; Gaidamaka, Y.; Samouylov, K. Multi-tenant resource sharing with equitable-priority-based performance isolation of slices for 5G cellular systems. *Comput. Commun.* **2022**, *188*, 39–51. [CrossRef]
182. Yang, W.; Cai, L.; Shu, S.; Pan, J. Mobility-aware congestion control for multipath QUIC in integrated terrestrial satellite networks. *IEEE Trans. Mob. Comput.* **2024**, *23*, 11620–11634. [CrossRef]
183. Chen, Y.; Liu, Z.; Zhang, Y.; Wu, Y.; Chen, X.; Zhao, L. Deep reinforcement learning-based dynamic resource management for mobile edge computing in industrial internet of things. *IEEE Trans. Ind. Inform.* **2020**, *17*, 4925–4934. [CrossRef]
184. Choudhary, A.; Kang, S.S.; Singla, S. A Comprehensive Analysis on Load Balancing of Software Defined Networking using Resource Optimization on AI-Based Applications. In Proceedings of the 2024 International Conference on Advances in Computing, Communication and Applied Informatics (ACCAI), Chennai, India, 21–22 March 2024; pp. 1–8.
185. Zhang, J.; Liu, Y.; Qin, X.; Xu, X.; Zhang, P. Adaptive resource allocation for blockchain-based federated learning in Internet of Things. *IEEE Internet Things J.* **2023**, *10*, 10621–10635. [CrossRef]
186. Di Lorenzo, P.; Battiloro, C.; Merluzzi, M.; Barbarossa, S. Dynamic resource optimization for adaptive federated learning at the wireless network edge. In Proceedings of the ICASSP 2021–2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Toronto, ON, Canada, 6–11 June 2021; pp. 4910–4914.
187. Zhang, Q.; Gui, L.; Hou, F.; Chen, J.; Zhu, S.; Tian, F. Dynamic task offloading and resource allocation for mobile-edge computing in dense cloud RAN. *IEEE Internet Things J.* **2020**, *7*, 3282–3299. [CrossRef]
188. Tang, J.; Jalalzai, M.M.; Feng, C.; Xiong, Z.; Zhang, Y. Latency-aware task scheduling in software-defined edge and cloud computing with erasure-coded storage systems. *IEEE Trans. Cloud Comput.* **2022**, *11*, 1575–1590. [CrossRef]
189. Xie, B.; Cui, H. Deep Reinforcement Learning for Task Offloading in Edge Computing. In Proceedings of the 2024 4th International Conference on Machine Learning and Intelligent Systems Engineering (MLISE), Zhuhai, China, 28–30 June 2024; pp. 250–254.
190. Hu, Z.; Niu, J.; Ren, T.; Guizani, M. Achieving fast environment adaptation of drl-based computation offloading in mobile edge computing. *IEEE Trans. Mob. Comput.* **2023**, *23*, 6347–6362. [CrossRef]
191. Zhang, S.; Gu, H.; Chi, K.; Huang, L.; Yu, K.; Mumtaz, S. DRL-based partial offloading for maximizing sum computation rate of wireless powered mobile edge computing network. *IEEE Trans. Wirel. Commun.* **2022**, *21*, 10934–10948. [CrossRef]
192. Ben Ammar, M.; Ben Dhaou, I.; El Houssaini, D.; Sahnoun, S.; Fakhfakh, A.; Kanoun, O. Requirements for energy-harvesting-driven edge devices using task-offloading approaches. *Electronics* **2022**, *11*, 383. [CrossRef]
193. Chen, H.; Qin, W.; Wang, L. Task partitioning and offloading in IoT cloud-edge collaborative computing framework: A survey. *J. Cloud Comput.* **2022**, *11*, 86. [CrossRef]
194. Suganya, B.; Gopi, R.; Kumar, A.R.; Singh, G. Dynamic task offloading edge-aware optimization framework for enhanced UAV operations on edge computing platform. *Sci. Rep.* **2024**, *14*, 16383. [CrossRef]
195. Farahbakhsh, F.; Shahidinejad, A.; Ghobaei-Arani, M. Context-aware computation offloading for mobile edge computing. *J. Ambient Intell. Humaniz. Comput.* **2023**, *14*, 5123–5135. [CrossRef]
196. Zhang, Q.; Gui, L.; Zhu, S.; Lang, X. Task offloading and resource scheduling in hybrid edge-cloud networks. *IEEE Access* **2021**, *9*, 85350–85366. [CrossRef]

197. Li, H.; Li, X.; Sun, C.; Fang, F.; Fan, Q.; Wang, X.; Leung, V.C. Intelligent content caching and user association in mobile edge computing networks for smart cities. *IEEE Trans. Netw. Sci. Eng.* **2023**, *11*, 994–1007. [CrossRef]
198. Ullah, I.; Khan, M.S.; St-Hilaire, M.; Faisal, M.; Kim, J.; Kim, S.M. Task Priority-Based Cached-Data Prefetching and Eviction Mechanisms for Performance Optimization of Edge Computing Clusters. *Secur. Commun. Netw.* **2021**, *2021*, 5541974. [CrossRef]
199. Yang, Z.; Fu, Y.; Liu, Y.; Chen, Y.; Zhang, J. A new look at AI-driven NOMA-F-RANs: Features extraction, cooperative caching, and cache-aided computing. *IEEE Wirel. Commun.* **2022**, *29*, 123–130. [CrossRef]
200. Maher, S.M.; Ebrahim, G.A.; Hosny, S.; Salah, M.M. A cache-enabled device-to-device approach based on deep learning. *IEEE Access* **2023**, *11*, 76953–76963. [CrossRef]
201. Weerasinghe, S.; Zaslavsky, A.; Loke, S.W.; Hassani, A.; Abken, A.; Medvedev, A. From traditional adaptive data caching to adaptive context caching: A survey. *arXiv* **2022**, arXiv:2211.11259.
202. Tolba, B.; Abo-Zahhad, M.; Elsabrouty, M.; Uchiyama, A.; Abd El-Malek, A.H. Joint user association, service caching, and task offloading in multi-tier communication/multi-tier edge computing heterogeneous networks. *Ad Hoc Netw.* **2024**, *160*, 103500. [CrossRef]
203. Babou, C.S.M.; Fall, D.; Kashihara, S.; Taenaka, Y.; Bhuyan, M.H.; Niang, I.; Kadobayashi, Y. Hierarchical load balancing and clustering technique for home edge computing. *IEEE Access* **2020**, *8*, 127593–127607. [CrossRef]
204. Chen, H.; Liu, J. Burst load scheduling latency optimization through collaborative content caching in edge-cloud computing. *Clust. Comput.* **2025**, *28*, 166. [CrossRef]
205. Zhang, S.; Gao, Q.; Wu, H. Cooperative Edge Caching with Multi-Agent Reinforcement Learning Using Consensus Updates. In Proceedings of the 2024 IEEE 7th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), Chongqing China, 20–22 September 2024; Volume 7, pp. 1804–1809.
206. Liang, Q.; Hanafy, W.A.; Ali-Eldin, A.; Shenoy, P. Model-driven cluster resource management for ai workloads in edge clouds. *ACM Trans. Auton. Adapt. Syst.* **2023**, *18*, 1–26. [CrossRef]
207. Xia, Q.; Ren, W.; Xu, Z.; Wang, X.; Liang, W. When edge caching meets a budget: Near optimal service delivery in multi-tiered edge clouds. *IEEE Trans. Serv. Comput.* **2021**, *15*, 3634–3648. [CrossRef]
208. Xing, H.; Ding, Y.; Huang, H.; Chen, Z.; Liu, S.; Guo, Z.; Al-Hasan, M.; Serhani, M.A.; Xu, Y. Hierarchical Sketch: An Efficient, Scalable and Latency-aware Content Caching Design for Content Delivery Networks. In Proceedings of the 2024 IEEE/ACM 32nd International Symposium on Quality of Service (IWQoS), Guangzhou, China, 19–21 June 2024; pp. 1–6.
209. Maheshwari, S.; Raychaudhuri, D.; Seskar, I.; Bronzino, F. Scalability and performance evaluation of edge cloud systems for latency constrained applications. In Proceedings of the 2018 IEEE/ACM Symposium on Edge Computing (SEC), Seattle, WA, USA, 25–27 October 2018; pp. 286–299.
210. Popovski, P.; Stefanović, Č.; Nielsen, J.J.; De Carvalho, E.; Angjelichinoski, M.; Trillingsgaard, K.F.; Bana, A.S. Wireless access in ultra-reliable low-latency communication (URLLC). *IEEE Trans. Commun.* **2019**, *67*, 5783–5801. [CrossRef]
211. He, Q.; Dong, Z.; Chen, F.; Deng, S.; Liang, W.; Yang, Y. Pyramid: Enabling hierarchical neural networks with edge computing. In Proceedings of the ACM Web Conference 2022, Lyon, France 25–29 April 2022; pp. 1860–1870.
212. Zhao, Y.; Liu, X.; Tu, L.; Tian, C.; Qiao, C. Dynamic service entity placement for latency sensitive applications in transportation systems. *IEEE Trans. Mob. Comput.* **2019**, *20*, 460–472. [CrossRef]
213. Ouyang, T.; Zhou, Z.; Chen, X. Follow me at the edge: Mobility-aware dynamic service placement for mobile edge computing. *IEEE J. Sel. Areas Commun.* **2018**, *36*, 2333–2345. [CrossRef]
214. Anwar, M.R.; Wang, S.; Akram, M.F.; Raza, S.; Mahmood, S. 5G-enabled MEC: A distributed traffic steering for seamless service migration of internet of vehicles. *IEEE Internet Things J.* **2021**, *9*, 648–661. [CrossRef]
215. Gür, G.; Kalla, A.; De Alwis, C.; Pham, Q.V.; Ngo, K.H.; Liyanage, M.; Porambage, P. Integration of ICN and MEC in 5G and beyond networks: Mutual benefits, use cases, challenges, standardization, and future research. *IEEE Open J. Commun. Soc.* **2022**, *3*, 1382–1412. [CrossRef]
216. Xu, Z.; Zhang, Y.; Li, H.; Yang, W.; Qi, Q. Dynamic resource provisioning for cyber-physical systems in cloud-fog-edge computing. *J. Cloud Comput.* **2020**, *9*, 32. [CrossRef]
217. Rasheed, Z.; Ma, Y.K.; Ullah, I.; Tao, Y.; Khan, I.; Khan, H.; Shafiq, M. Edge Computing in the Digital Era: The Nexus of 5G, IoT and a Seamless Digital Future. In *Future Communication Systems Using Artificial Intelligence, Internet of Things and Data Science*; CRC Press: Boca Raton, FL, USA, 2024; pp. 213–234.
218. Jararweh, Y. Enabling efficient and secure energy cloud using edge computing and 5G. *J. Parallel Distrib. Comput.* **2020**, *145*, 42–49. [CrossRef]
219. Darwich, M.; Khalil, K.; Bayoumi, M. Adaptive Multi-Path Video Streaming Using AI-Driven Edge Computing for Enhanced Quality of Experience (QoE). In Proceedings of the 2024 2nd International Conference on Artificial Intelligence, Blockchain, and Internet of Things (AIBThings), Mt Pleasant, MI, USA, 7–8 September 2024; pp. 1–6.
220. Bulej, L.; Bureš, T.; Filandr, A.; Hnětynka, P.; Hnětynková, I.; Pacovský, J.; Sandor, G.; Gerostathopoulos, I. Managing latency in edge–cloud environment. *J. Syst. Softw.* **2021**, *172*, 110872. [CrossRef]

221. Xiao, Y.; Jia, Y.; Liu, C.; Cheng, X.; Yu, J.; Lv, W. Edge computing security: State of the art and challenges. *Proc. IEEE* **2019**, *107*, 1608–1631. [CrossRef]
222. Bonnah, E.; Shiguang, J. DecChain: A decentralized security approach in Edge Computing based on Blockchain. *Future Gener. Comput. Syst.* **2020**, *113*, 363–379. [CrossRef]
223. Ma, Y.; Liu, L.; Liu, Z.; Li, F.; Xie, Q.; Chen, K.; Lv, C.; He, Y.; Li, F. A Survey of DDoS Attack and Defense Technologies in Multi-Access Edge Computing. *IEEE Internet Things J.* **2024**, 12, 1428–1452. [CrossRef]
224. Zhou, H.; Yang, G.; Dai, H.; Liu, G. PFLF: Privacy-preserving federated learning framework for edge computing. *IEEE Trans. Inf. Forensics Secur.* **2022**, *17*, 1905–1918. [CrossRef]
225. Alluhaidan, A.S.D.; Prabu, P. End-to-end encryption in resource-constrained IoT device. *IEEE Access* **2023**, *11*, 70040–70051. [CrossRef]
226. Kolevski, D.; Michael, K. Edge Computing and IoT Data Breaches: Security, Privacy, Trust, and Regulation. *IEEE Technol. Soc. Mag.* **2024**, *43*, 22–32. [CrossRef]
227. Ruan, L.; Guo, S.; Qiu, X.; Meng, L.; Wu, S.; Buyya, R. Edge in-network computing meets blockchain: A multi-domain heterogeneous resource trust management architecture. *IEEE Netw.* **2021**, *35*, 50–57. [CrossRef]
228. Li, C.; Liang, S.; Zhang, J.; Wang, Q.-E.; Luo, Y. Blockchain-based data trading in edge-cloud computing environment. *Inf. Process. Manag.* **2022**, *59*, 102786. [CrossRef]
229. Haque, E.U.; Shah, A.; Iqbal, J.; Ullah, S.S.; Alroobaea, R.; Hussain, S. A scalable blockchain based framework for efficient IoT data management using lightweight consensus. *Sci. Rep.* **2024**, *14*, 7841.
230. Zhong, J.; Wu, C.; Liu, D.; Shen, Z.; Wang, X. Intelligent IoT Device Abnormal Traffic Detection Method Based on Secure Multi-Party Computation. In Proceedings of the 2024 3rd Asian Conference on Frontiers of Power and Energy (ACFPE), Chengdu, China, 25–27 October 2024; pp. 331–335.
231. Liu, S.; Wang, X.; Hui, L.; Wu, W. Blockchain-based decentralized federated learning method in edge computing environment. *Appl. Sci.* **2023**, *13*, 1677. [CrossRef]
232. Yang, Q. Toward responsible ai: An overview of federated learning for user-centered privacy-preserving computing. *ACM Trans. Interact. Intell. Syst. (TiiS)* **2021**, *11*, 1–22. [CrossRef]
233. Shahidinejad, A.; Ghobaei-Arani, M.; Souri, A.; Shojafar, M.; Kumari, S. Light-edge: A lightweight authentication protocol for IoT devices in an edge-cloud environment. *IEEE Consum. Electron. Mag.* **2021**, *11*, 57–63. [CrossRef]
234. Shen, S.; Ren, Y.; Ju, Y.; Wang, X.; Wang, W.; Leung, V.C. Edgematrix: A resource-redefined scheduling framework for sla-guaranteed multi-tier edge-cloud computing systems. *IEEE J. Sel. Areas Commun.* **2022**, *41*, 820–834. [CrossRef]
235. Guim, F.; Metsch, T.; Moustafa, H.; Verrall, T.; Carrera, D.; Cadenelli, N.; Chen, J.; Doria, D.; Ghadie, C.; Prats, R.G. Autonomous lifecycle management for resource-efficient workload orchestration for green edge computing. *IEEE Trans. Green Commun. Netw.* **2021**, *6*, 571–582. [CrossRef]
236. Huang, M.; Li, Z.; Xiao, F.; Long, S.; Liu, A. Trust mechanism-based multi-tier computing system for service-oriented edge-cloud networks. *IEEE Trans. Dependable Secur. Comput.* **2023**, *21*, 1639–1651. [CrossRef]
237. Tang, Z.; Jia, W.; Zhou, X.; Yang, W.; You, Y. Representation and reinforcement learning for task scheduling in edge computing. *IEEE Trans. Big Data* **2020**, *8*, 795–808. [CrossRef]
238. Zhang, P.; Chen, N.; Xu, G.; Kumar, N.; Barnawi, A.; Guizani, M.; Duan, Y.; Yu, K. Multi-target-aware dynamic resource scheduling for cloud-fog-edge multi-tier computing network. *IEEE Trans. Intell. Transp. Syst.* **2023**, *25*, 3885–3897. [CrossRef]
239. Qadeer, A.; Lee, M.J. Deep-deterministic policy gradient based multi-resource allocation in edge-cloud system: A distributed approach. *IEEE Access* **2023**, *11*, 20381–20398. [CrossRef]
240. Dreibholz, T.; Mazumdar, S. Towards a lightweight task scheduling framework for cloud and edge platform. *Internet Things* **2023**, *21*, 100651. [CrossRef]
241. Liu, J.; Xu, Z.; Wang, C.; Liu, X.; Xie, X.; Shi, G. Mobility-aware MEC planning with a GNN-based graph partitioning framework. *IEEE Trans. Netw. Serv. Manag.* **2024**, *21*, 4383–4395. [CrossRef]
242. Wu, C.L.; Chiu, T.C.; Wang, C.Y.; Pang, A.C. Mobility-aware deep reinforcement learning with seq2seq mobility prediction for offloading and allocation in edge computing. *IEEE Trans. Mob. Comput.* **2023**, *23*, 6803–6819. [CrossRef]
243. Lan, D.; Taherkordi, A.; Eliassen, F.; Chen, Z.; Liu, L. Deep reinforcement learning for intelligent migration of fog services in smart cities. In Proceedings of the Algorithms and Architectures for Parallel Processing: 20th International Conference, ICA3PP 2020, New York, NY, USA, 2–4 October 2020; Proceedings, Part II 20; Springer: Berlin/Heidelberg, Germany, 2020; pp. 230–244.
244. Le, T.H.T.; Tran, N.H.; LeAnh, T.; Oo, T.Z.; Kim, K.; Ren, S.; Hong, C.S. Auction mechanism for dynamic bandwidth allocation in multi-tenant edge computing. *IEEE Trans. Veh. Technol.* **2020**, *69*, 15162–15176. [CrossRef]
245. Tang, J.; Nie, J.; Xiong, Z.; Zhao, J.; Zhang, Y.; Niyato, D. Slicing-based reliable resource orchestration for secure software-defined edge-cloud computing systems. *IEEE Internet Things J.* **2021**, *9*, 2637–2648. [CrossRef]
246. Wu, Y. Cloud-edge orchestration for the Internet of Things: Architecture and AI-powered data processing. *IEEE Internet Things J.* **2020**, *8*, 12792–12805. [CrossRef]

247. Gong, Y.; Yao, H.; Wang, J.; Wu, D.; Zhang, N.; Yu, F.R. Decentralized edge intelligence-driven network resource orchestration mechanism. *IEEE Netw.* **2022**, *37*, 270–276. [CrossRef]
248. Fu, K.; Zhang, W.; Chen, Q.; Zeng, D.; Guo, M. Adaptive resource efficient microservice deployment in cloud-edge continuum. *IEEE Trans. Parallel Distrib. Syst.* **2021**, *33*, 1825–1840. [CrossRef]
249. Shen, X.; Yu, H.; Liu, X.; Bin, Q.; Luhach, A.K.; Saravanan, V. The optimized energy-efficient sensible edge processing model for the internet of vehicles in smart cities. *Sustain. Energy Technol. Assessments* **2021**, *47*, 101477. [CrossRef]
250. Cao, K.; Hu, S.; Shi, Y.; Colombo, A.W.; Karnouskos, S.; Li, X. A survey on edge and edge-cloud computing assisted cyber-physical systems. *IEEE Trans. Ind. Inform.* **2021**, *17*, 7806–7819. [CrossRef]
251. Zhao, Y.; Hu, N.; Zhao, Y.; Zhu, Z. A secure and flexible edge computing scheme for AI-driven industrial IoT. *Clust. Comput.* **2023**, *26*, 283–301. [CrossRef]
252. Zhiwei, Q.; Juan, L.; Xiao, L.; Mengyuan, Z. Energy-aware workflow real-time scheduling strategy for device-edge-cloud collaborative computing. *Comput. Integr. Manuf. Syst.* **2022**, *28*, 3122.
253. Bellal, Z.; Lahlou, L.; Kara, N.; El Khayat, I. Gas: Dvfs-driven energy efficiency approach for latency-guaranteed edge computing microservices. *IEEE Trans. Green Commun. Netw.* **2024**, *9*, 108–124. [CrossRef]
254. Wang, Y.; Zhang, W.; Hao, M.; Wang, Z. Online power management for multi-cores: A reinforcement learning based approach. *IEEE Trans. Parallel Distrib. Syst.* **2021**, *33*, 751–764. [CrossRef]
255. Patra, S.S.; Govindaraj, R.; Chowdhury, S.; Shah, M.A.; Patro, R.; Rout, S. Energy efficient end device aware solution through SDN in edge-cloud platform. *IEEE Access* **2022**, *10*, 115192–115204. [CrossRef]
256. Liu, W.; Zhang, H.; Ding, H.; Yuan, D. Delay and energy minimization for adaptive video streaming: A joint edge caching, computing and power allocation approach. *IEEE Trans. Veh. Technol.* **2022**, *71*, 9602–9612. [CrossRef]
257. Wang, R.; Friderikos, V.; Aghvami, A.H. Energy-aware design policy for network slicing using deep reinforcement learning. *IEEE Trans. Serv. Comput.* **2024**, 17, 2378–2391. [CrossRef]
258. Liu, X.; Yang, J.; Zou, C.; Chen, Q.; Yan, X.; Chen, Y.; Cai, C. Collaborative edge computing with FPGA-based CNN accelerators for energy-efficient and time-aware face tracking system. *IEEE Trans. Comput. Soc. Syst.* **2021**, *9*, 252–266. [CrossRef]
259. Danopoulos, D. Hardware-Software Co-Design of Deep Learning Accelerators: From Custom to Automated Design Methodologies. Ph.D. Thesis, National Technical University of Athens, Athens, Greece, 2024.
260. Silva, J.; Marques, E.R.; Lopes, L.M.; Silva, F. Energy-aware adaptive offloading of soft real-time jobs in mobile edge clouds. *J. Cloud Comput.* **2021**, *10*, 38. [CrossRef]
261. Luo, T.; Wong, W.F.; Goh, R.S.M.; Do, A.T.; Chen, Z.; Li, H.; Jiang, W.; Yau, W. Achieving green ai with energy-efficient deep learning using neuromorphic computing. *Commun. ACM* **2023**, *66*, 52–57. [CrossRef]
262. Abbasi, M.H.A.; Arshed, J.U.; Ahmad, I.; Afzal, M.; Ali, H.; Hussain, G. A Mobility Prediction Based Adaptive Task Migration in Mobile Edge Computing. *VFAST Trans. Softw. Eng.* **2024**, *12*, 46–55.
263. Tuli, S.; Casale, G.; Jennings, N.R. Pregan: Preemptive migration prediction network for proactive fault-tolerant edge computing. In Proceedings of the IEEE INFOCOM 2022-IEEE Conference on Computer Communications, London, UK, 2–5 May 2022; pp. 670–679.
264. Shahzadi, S.; Iqbal, M.; Dagiuklas, T.; Qayyum, Z.U. Multi-access edge computing: Open issues, challenges and future perspectives. *J. Cloud Comput.* **2017**, *6*, 30. [CrossRef]
265. Aly, M.; Khomh, F.; Guéhéneuc, Y.G.; Washizaki, H.; Yacout, S. Is Fragmentation a Threat to the Success of the Internet of Things? *IEEE Internet Things J.* **2018**, *6*, 472–487. [CrossRef]
266. Ning, H.; Li, Y.; Shi, F.; Yang, L.T. Heterogeneous edge computing open platforms and tools for internet of things. *Future Gener. Comput. Syst.* **2020**, *106*, 67–76. [CrossRef]
267. Gupta, R.; Nair, A.; Tanwar, S.; Kumar, N. Blockchain-assisted secure UAV communication in 6G environment: Architecture, opportunities, and challenges. *IET Commun.* **2021**, *15*, 1352–1367. [CrossRef]
268. Vankayalapati, R.K. Unifying Edge and Cloud Computing: A Framework for Distributed AI and Real-Time Processing. *SSRN* **2023**. [CrossRef]
269. Salako, A.; Fabuyi, J.; Aideyan, N.T.; Selesi-Aina, O.; Dapo-Oyewole, D.L.; Olaniyi, O.O. Advancing Information Governance in AI-Driven Cloud Ecosystem: Strategies for Enhancing Data Security and Meeting Regulatory Compliance. *SSRN* **2024**. [CrossRef]
270. Ullah, R.; Rehman, M.A.U.; Kim, B.S. Design and implementation of an open source framework and prototype for named data networking-based edge cloud computing system. *IEEE Access* **2019**, *7*, 57741–57759. [CrossRef]
271. Filip, I.D.; Postoaca, A.V.; Stochitoiu, R.D.; Neatu, D.F.; Negru, C.; Pop, F. Data capsule: Representation of heterogeneous data in cloud-edge computing. *IEEE Access* **2019**, *7*, 49558–49567. [CrossRef]
272. Merlino, G.; Dautov, R.; Distefano, S.; Bruneo, D. Enabling workload engineering in edge, fog, and cloud computing through OpenStack-based middleware. *ACM Trans. Internet Technol. (TOIT)* **2019**, *19*, 1–22. [CrossRef]
273. Kreković, D.; Krivić, P.; Žarko, I.P.; Kušek, M.; Le-Phuoc, D. Reducing Communication Overhead in the IoT-Edge-Cloud Continuum: A Survey on Protocols and Data Reduction Strategies. *arXiv* **2024**, arXiv:2404.19492. [CrossRef]

274. Veeramachaneni, V. Edge Computing: Architecture, Applications, and Future Challenges in a Decentralized Era. *Recent Trends Comput. Graph. Multimed. Technol.* **2025**, *7*, 8–23.
275. Ahmed, A.; Azizi, S.; Zeebaree, S.R. ECQ: An energy-efficient, cost-effective and qos-aware method for dynamic service migration in mobile edge computing systems. *Wirel. Pers. Commun.* **2023**, *133*, 2467–2501. [CrossRef]