Sensors and Materials, Vol. 38, No. 2 (2026) 985–999
MYU Tokyo

S & M 4359

Real-time Data Processing Optimization in Smart City Using Internet of Things and Sensor Network

Yan Hu,[1] Yi Pan,[1] and Zengtao Wu[2*]

[1]College of Intelligent Transportation, Hunan Communication Polytechnic, Changsha 410132, China
[2]Highway Professional Qualification Division of the Professional Qualification Authority of the Ministry of Transport, Beijing 100029, China

(Received December 21, 2025; accepted February 6, 2026)

Keywords: smart cities, IoT, real-time data processing, edge computing, stream processing

Smart cities have been developed by integrating IoT networks that generate huge data from heterogeneous sources, including environmental sensors, surveillance systems, and utility meters. Processing this vast volume of data faces challenges regarding nonstandard device protocols, limited network capacity, and the need for subsecond latency in critical tasks like autonomous vehicle coordination. Therefore, appropriate strategies are required for optimizing real-time sensor data processing through the integration of edge and cloud computing with advanced analytics. In this study, we aim to review optimization technologies, including adaptive sampling. It is critical to adjust collection rates while considering anomalies and construct tiered data storage architectures that separate hot, warm, and cold data for efficiency. Furthermore, reinforcement learning and container orchestration platforms such as Kubernetes must be introduced for dynamic resource distribution and load balancing across the computing continuum. The practical application of these architectures in Shenzhen, China, is highlighted. Shenzhen's City Brain (named after the Hangzhou City Brain), utilizing distributed edge-cloud processing, reduced average traffic intersection waiting times by 12.6% and achieved a 92% detection rate for environmental anomalies using fog-computing gateways. Transforms from batch-oriented processes to real-time methods led to significant changes in urban governance. However, sustaining these systems requires addressing security vulnerabilities, ensuring data privacy through differential privacy, and standardizing interoperability policies to maximize the return on investment for smart cities.

1. Introduction

The concept of smart cities extends beyond computerization, heavily relying on the macro-level integration of IoT networks.(1) Distributed networks for decision-making enhance municipal operations, thereby improving the quality of life for citizens. By 2050, an estimated 68% of the global population will dwell in urban areas, necessitating prudent and responsive networks with efficiently distributed infrastructure to sustain and deliver high-quality services.(2) Smart cities

*Corresponding author: e-mail: zengtaowu@foxmail.com
https://doi.org/10.18494/SAM6140

ISSN 0914-4935

require sensor networks, actuators, and computational nodes that are integrated in cyber-physical systems. These systems continuously monitor, analyze, and optimize transport infrastructure, energy distribution, environmental regulation, and urban security (Fig. 1).[3] Their operations must rapidly respond, utilizing the advanced data processing of dynamic conditions in smart cities. The ability to stream data from millions of IoT devices with minimal latency in operation enables adaptive traffic management, proactive maintenance, and effective emergency response systems.

Smart city infrastructures generate exabytes of data from heterogeneous sources, including environmental sensors, surveillance systems, utility meters, and mobile devices. The essential concept of the smart city infrastructure is IoT, in which vast sensor networks collect data in the sensing layer of the smart city archtecture.[4] This data is heterogeneous owing to the diverse technologies and applications involved.[5] The smart city infrastructure requires diverse sensor technologies and generates massive amounts of data. Environmental sensors are used to monitor localized conditions and maintain public health and sustainability. Air quality, temperature, humidity, and noise, and other meteorological data are collected and analyzed.[4] Surveillance systems are employed to provide high-resolution visual and spatial data used for safety, security, and detailed infrastructure analysis. In the surveillance system, CMOS image sensors are used for standard 2D/3D video monitoring, light detection and ranging (LiDAR) systems are used for high-precision 3D mapping and object detection, and advanced event-based sensors are deployed

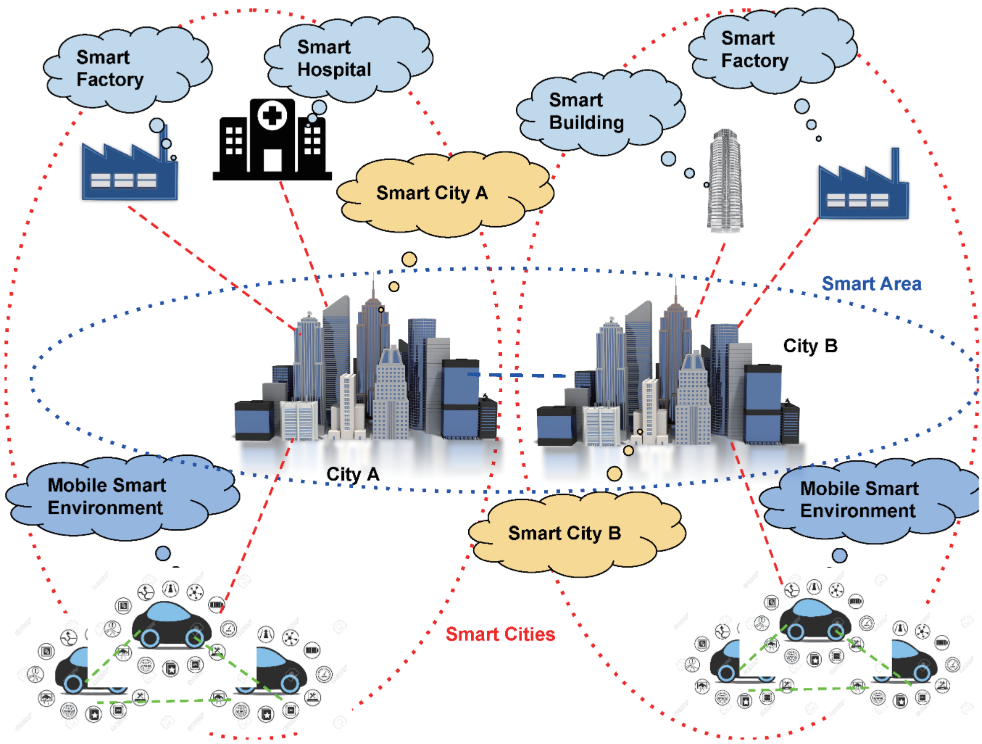

Fig. 1. (Color online) Example of smart city system.[3]

for movement capture and change with minimal bandwidth.[6] For efficient resource management, smart utility meters are used to monitor electric and gas consumption for the proactive management of the grid and the prompt detection of leaks.[6] Mobile devices function as participatory sensors in the architecture. GPS sensors and accelerometers in the devices are used for tracking movement and traffic flow. Vehicle-mounted sensors are used to determine vehicle location and assess traffic flow rates, parking availability, and pedestrian movement for traffic control.[7]

Processing the vast data collected requires sophisticated architectures that ingest, analyze, and produce actionable decision-making. The results lead to fundamental changes in urban governance: decision-making is changing from retrospective, batch-oriented processes to prospective, real-time methods. Such a change allows AI systems to efficiently prevent infrastructure failures and adverse events, and allocate resources. Despite such advancements, deployments of data stream and analysis processes face the challenges of nonstandard device protocols, limited network capacity, computational constraints, and data integrity. Addressing these challenges requires delivering subsecond latency for time-critical tasks such as autonomous vehicle coordination and emergency service dispatch.

In this study, we examine strategies for optimizing real-time sensor data processing architectures in smart cities, especially on the integration of edge and cloud computing and advanced analytics. First, we characterize architectural elements and data-flow models of smart cities, and compare data-stream-processing frameworks with batch-processing architectures. We also analyze optimization algorithms used for resource distribution and computational load placement in smart city implementations to forecast future technological trends.

2. Literature Review

Real-time data processing in smart cities using IoT has been extensively researched, especially regarding architectural design, algorithmic optimization, and empirical deployment studies. Li *et al.* investigated big data analysis in IoT and optimized latency-conscious service provisioning in a hierarchical architecture comprising edge, fog, and cloud layers.[8] Deep-learning-based urban big data fusion and real-time data processing are used for traffic monitoring by utilizing convolutional neural networks deployed at the network edge.[9] In addition, real-time big data analytics enhances urban planning by leveraging sensors to generate continuous, asynchronous data streams requiring complex time-coordination mechanisms.[10] Integrating stream processing in the IoT sensor network is essential for improving performance and decision-making in smart city architectures. Zanella *et al.* defined the IoT-based smart city architecture using a web service approach (Fig. 2).[11]

Figure 2 shows the communication architecture of a modern smart city. In a large city, thousands of devices, including air quality sensors, smart streetlights, and water meters, send information to a central control center. Because these small devices have limited battery power and memory, they cannot use the same communication protocols that a laptop computer or a server uses. The architecture in Fig. 2 acts like a language translation system, allowing simple, low-power sensors to communicate with the powerful global Internet.

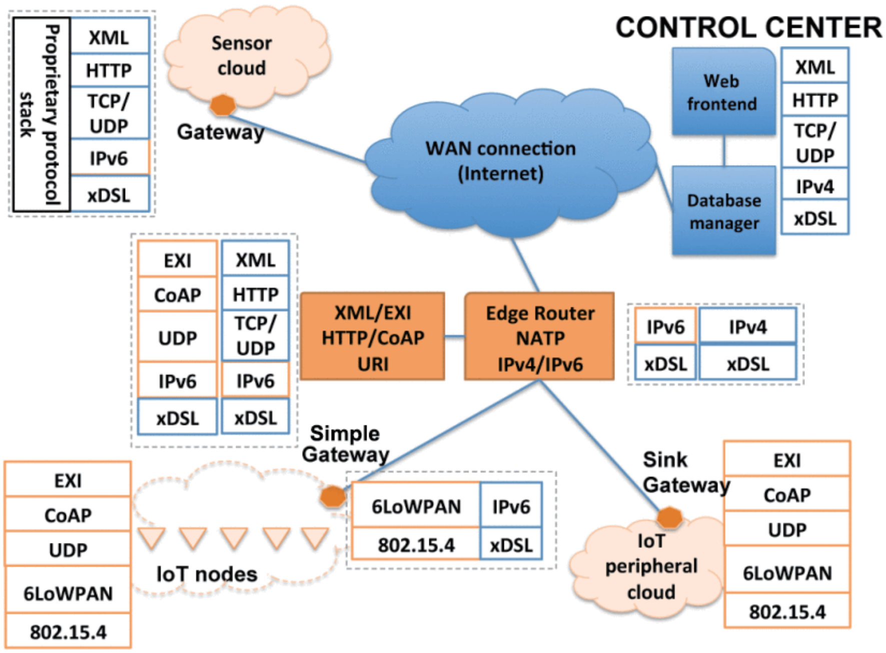

Fig. 2. (Color online) Urban IoT network based on web service in smart city architecture.(11) (6LoWPAN: IPV6 over low-power wireless personal area networks, 802.15.4: IEEE 802.15.4, CoAP: constrained application protocol, EXI: efficient XML interchange, HTTP: hypertext transfer protocol, IPV4: Internet Protocol Version 4, IPV6: Internet Protocol Version 6, NATP: network address and port translation, TCP: transmission control protocol, UDP: user datagram protocol, URI: uniform resource identifier, XDSL: digital subscriber line, and XML: extensible markup language).

These devices are the physical sensors scattered throughout the city. They use lightweight protocols such as 6LoWPAN and 802.15.4, which are designed to transmit a small amount of data while consuming almost no battery power. Such a digital shorthand is efficient but operates in short distances. Because the digital shorthand used by sensors cannot be understood by the standard Internet, gateways and edge routers act as intermediaries. They receive the simple signals from the sensors and translate the data into standard Internet protocols such as IPv6 and HTTP.

The edge router handles the transition from a local city network to the global wide area network (WAN). WAN is the standard global network. Once the data reaches this stage, it is formatted as a piece of web traffic using TCP/UDP and IPv4/IPv6, enabling it to travel long distances to reach a central server. The control center processes and stores the data for long-term analysis. A user-friendly dashboard on a website allows managers to view real-time maps, alerts, and statistics about the city's health and infrastructure.

In smart cities, technological infrastructures comprise IoT devices, sensor networks, and communication protocols across multiple architectural levels. Performance-oriented IoT applications are employed with sensors to provide data sources under the low-power wide-area network protocols in the long-range wide area network as the IoT computing continuum (IoTinuum) (Fig. 3).[12] Mrabet *et al.* proposed a five-layer IoT architecture encompassing

Fig. 3. (Color online) IoT computing continuum (IoTinuum).(12)

physical perception, network and protocol, transport, application, and data service layers, each integrated with security mechanisms.(13) Tekinerdogan *et al.* analyzed IoT-based smart city system architectures to collect sensor data from the devices that are transmitted to network layers.(14) Middleware layers are added to enhance the architecture by offering appropriate techniques and common functionality, while the data collected are processed in the four stages of sensing, networking, analysis, and implementation.(15)

Software-defined networking combined with network functions enables the dynamic allocation of network resources, adjusting bandwidth and routing to accommodate fluctuating urban IoT data. Real-time analytical technologies are critical for processing such fluctuating data in the smart city architecture. Apache Kafka (distributed message brokering), Apache Flink (stateful stream processing with event-time semantics), and Apache Storm (real-time computation and ingestion) have exhibited distinct architectural designs and performance for different operational requirements. Apache Storm employs spout-bolt architectures to support low-latency, tuple-at-a-time processing with at-least-once semantics. Apache Flink integrates unified batch-stream processing through built-in streaming, offering low latency and robust fault tolerance. Apache Kafka, when coupled with Kafka brokers, supports record-at-a-time processing with the local state replicated to Kafka topics. Apache Flink and Apache Kafka demonstrate scalability and near-linear performance as additional cloud resources, compared with Apache Beam.(16) The compatibility of these frameworks is affected by different resource requirements under increasing workloads. Apache Flink, a declarative engine, automatically infers directed acyclic graphs (DAGs) from transformation ordering, whereas Apache Storm requires explicit DAG specifications, which enable it to provide fine-grained control but introduce additional implementation complexity. In the optimization of the smart city infrastructure, resource allocation and load balancing are vital. Therefore, reinforcement-learning-based strategies are often used.(16) Multi-observation single-state models based on reinforcement learning coordinate task allocation by leveraging task characteristics and executor states, and improve their performance through environmental learning processes.

The sixth generation of the smart city architecture, based on 6G-enabled technologies, enables intelligent task offloading and resource allocation.(17) The rapid expansion of Internet of Everything technologies and data volumes expected in the next generation smart city is necessary to provide energy management, intelligent transport, and e-health.(18) Su *et al.* proposed multi-objective optimization algorithms for virtual resource allocation in cloud load-balancing for smart city architecture.(19)

Throughout the development of smart city architectures, security, privacy, and interoperability are critical. Sampaio *et al.* developed pseudonymization and differential privacy mechanisms to safeguard individual privacy and data privacy.(20) In particular, human-centered smart cities must resist security threats, unauthorized data access, and potential biases in automated decision systems.(21) To address the diverse challenges in smart city architectures, it is essential to standardize device protocols and architectural frameworks, overcome scalability constraints, optimize energy consumption in battery-powered IoT devices for high-frequency sensing, and conduct longitudinal studies to evaluate system performance under real-world operating conditions. The results contribute to the advancement of theoretical models and practical implementations of smart city data processing, which is essential in the smart city architecture.

3. Real-time Data Processing

3.1 Perception layer and data acquisition

The smart city architecture with real-time data processing capabilities comprises interrelated components that enable data acquisition, transmission, processing, and actuation across hierarchical levels (Fig. 4). In the perception layer, diverse IoT devices serve as data sources. The devices include environmental sensors that monitor air quality parameters, the electrical conductivity of dissolved ions and pH in water, smart meters, traffic monitoring devices (inductive loop detectors, radar systems, and computer vision cameras), and surveillance systems that provide video feeds for security monitoring. These devices generate continuous data streams at various frequencies, ranging from millisecond-level accelerometer readings of

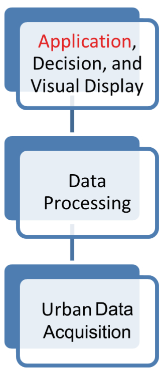

Fig. 4. (Color online) Smart city architecture with real-time data processing.

structural vibrations to hourly observations obtained from meteorological data, producing asynchronous data flows. Microcontrollers associated with the devices integrate communication units for localized data filtering, temporal aggregation, outlier detection, and lossy compression. These operations reduce network bandwidth requirements and energy consumption, extending the operational lifespan of battery-powered deployments.

3.2 Wireless data transmission protocols

Wireless data transmission in smart cities is tailored to application-specific requirements, including range, bandwidth, latency, power efficiency, and installation cost.

- Short-range protocols: Bluetooth Low Energy and Zigbee (IEEE 802.15.4), operating at 2.4 GHz with ranges up to 100 m. These are suited for building personal area networks (PAN), offering low power consumption and multiyear battery life but limited coverage.
- Medium-range protocols: IEEE 802.11 Wi-Fi standards provide higher bandwidth, supporting video streaming and high-frequency sensor arrays, though their energy demands limit battery-powered use.
- Long-range protocols: LoRaWAN (sub-GHz ISM bands, up to 10 km) and Narrowband IoT (licensed cellular spectrum) enable metropolitan-scale connectivity with minimal infrastructure. These support battery-powered devices lasting 5–10 years, transmitting at low data rates (0.3–50 kbps) for infrequent sensor updates (Fig. 5).[22]

3.3 5G network

Presently, 5G networks enhance smart city capabilities by providing ultrareliable, low-latency mobile communication with sub-millisecond round-trip times and data rates reaching hundreds

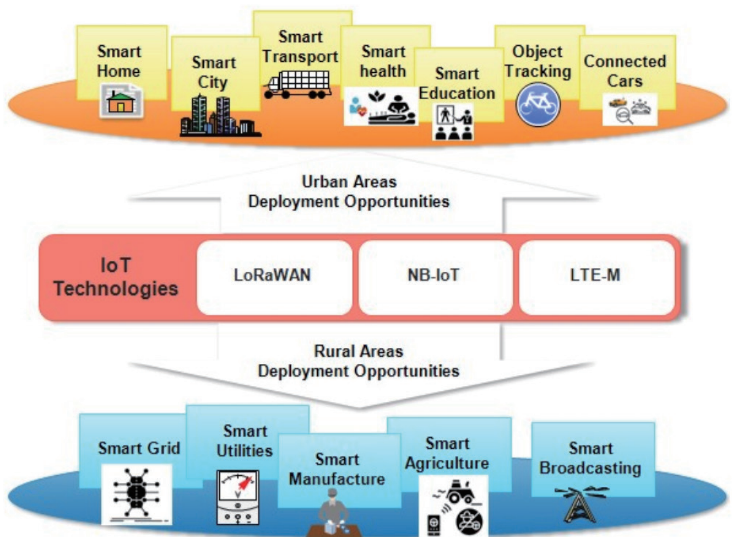

Fig. 5. (Color online) Metropolitan-scale connectivity with diverse infrastructure (NB-IoT: narrowband IoT, LTE-M: long-term evolution-machine).

of gigabytes per second. Such performance is critical for mission-oriented applications, including the following:

- Autonomous vehicle coordination requires vehicle-to-everything connectivity.
- Remote surgery requires tactile internet functionality.

Mesh topologies are commonly employed at network edges to enable single-hop routing with redundancy and self-recovery. Hierarchical aggregation points concentrate data flows before transmission to central infrastructures. Quality of service (QoS) mechanisms prioritize time-sensitive data streams using differentiated or integrated service models, ensuring critical alerts and control signals receive guaranteed bandwidth.

3.4 Data processing

Real-time data processing frameworks are designed according to latency requirements and optimization techniques.

- Stream processing: Apache Kafka, Flink, and Storm support incident detection and resource dispatch with latencies under one second. These applications maintain the computational state within tumbling, sliding, or session windows, enabling complex event processing by correlating spatiotemporal patterns across multiple streams to detect anomalies, identify trends, and trigger automated responses.
- Batch processing: Apache Hadoop MapReduce and Apache Spark operate on large, bounded datasets collected over defined intervals (minutes to hours). These frameworks support deep analytical capabilities, which are useful in urban planning where historical trend analysis informs predictive maintenance strategies.

Supervised learning models (random forests, gradient boosting machines, and deep neural networks) trained on historical data predict traffic congestion, energy demand, and infrastructure maintenance needs. Reinforcement learning enables adaptive systems to learn optimal control policies through iterative interactions with the environment, such as dynamically adjusting traffic signal timing to minimize travel times across networks.

3.5 Challenges and required implementation

The throughput of data stream processing is determined by parallelization capacity and coordination overhead, and is expressed in events per second for distributed clusters. However, integration with existing municipal infrastructure presents technical and organizational challenges, requiring sophisticated architectural planning and incremental implementation. Supervisory control and data acquisition networks and GIS, which are often used in the data stream process, rely on monolithic database models and heterogeneous technology stacks with limited interoperability. Therefore, they are implemented with intermediate layers to incorporate protocol translation, data transformation, and semantic mapping for communication among heterogeneous systems without the wholesale infrastructure. To extend their functionality, application programming interfaces (APIs) are adopted in representational state transfer architectures or message-oriented middleware designs. Data lakes with schema-on-read paradigms are also used to support diverse data formats from multiple sources, enabling unified analytics across siloed information systems. To adopt legacy systems, such as GIS, inter-institutional coordination for human resource management and procurement is required for incremental implementation strategies. These strategies enable satisfactory returns on investment for broad system deployments.

4. Data Optimization

4.1 Adaptive sampling and edge analytics

Optimization strategies for data collection and management minimize redundancy, improve data quality, and enable efficient storage that supports real-time access and long-term retention. Adaptive sampling techniques adjust the sampling rate corresponding to system features, anomalies, or state transitions, increasing sampling frequency. Edge analytics are used to verify data integrity at the sensor node by detecting outliers and features for transmission, eliminating abnormal data caused by sensor failures or environmental interference. Data compression contributes to efficient data processing and analysis. Lossy compression is applied to continuous sensor readings (e.g., color measurements) to reduce file size by discarding information that is less important for human perception, while lossless compression is used for discrete data, reducing bandwidth and storage requirements without compromising analytical utility by reducing file size while allowing the original data to be perfectly reconstructed from the compressed version. Delta encoding is used to record changes in values, being effective for monitoring gradually varying quantities such as temperature and pressure.

4.2 Data storage and time-series database

For data storage, tiered architectures are used to distinguish hot, warm, and cold data. Hot data require instant access and are stored in distributed in-memory databases such as Apache Ignite or Redis. Warm data are accessed periodically and stored in solid-state drive arrays, while cold data are archived in object storage systems such as Amazon S3 or Azure Blob Storage.

Time-series databases, such as InfluxDB and TimescaleDB, are optimized for IoT sensor data. They support efficient range queries, downsampling operations, and on-the-fly aggregate maintenance. Data lakes serve as scalable repositories of raw sensor data, enabling compliance activities and retrospective processing as new analytical methods emerge.[23]

4.3 Real-time analytics

It is important to reduce end-to-end latency between data generation and insight delivery while maintaining throughput and computational efficiency in real-time analytics. Algorithms that balance accuracy, computational complexity, and latency constraints must be selected. Lightweight statistical methods, such as moving averages, exponential smoothing, and percentile calculations, are mainly used for initial data filtering before applying computationally intensive machine learning models. Incremental learning algorithms are used to continuously update predictive models using newly acquired data, enabling adaptive control and responsiveness to evolving urban dynamics. Approximation algorithms, such as Count-Min Sketch for frequency estimation and HyperLog for cardinality estimation, are used to trade off accuracy with computational efficiency, allowing the analysis of large data streams with constant memory footprints.

Query optimization techniques, including predicate pushdown, source-level filtering, and materialized views, are used to reduce execution time. The stream-table join enriches real-time event data with reference data from dimension tables. It enables efficient vehicle trajectory analysis using GPS coordinates with road network topology. Temporal data streams are structured using stateful computations in windowing operations, such as tumbling, sliding, or session windows.(24) Event processing languages, including Structured Query Language extensions for streaming data, enable the declaration of multi-event spatiotemporal patterns, such as traffic congestion formation across multiple road segments.

4.4 Resource distribution and load balancing

An efficient resource distribution ensures the optimization of heterogeneous infrastructures that comprise edge devices, fog nodes, and cloud data centers. Dynamic task assignment algorithms are used to determine computational workloads by considering data locality, computational capacity, and latency responsiveness. Reinforcement learning, such as a Markov decision process, acquires optimal allocation policies depending on system dynamics.(17) Multi-objective optimization methods, such as Pareto-optimal solutions, are used to balance conflicting tasks, such as energy consumption and distribution.

Load balancing strategies are employed to allocate requests using round-robin to evenly distribute requests, least connections to direct requests to instances with fewer active connections, and consistent hashing to maintain client affinity with specific servers for sticky sessions. For efficient load balancing, container orchestration platforms (e.g., Kubernetes) are used to automate the deployment, scaling, and management of containerized applications. They provide service discovery, scaling, fault tolerance, and built-in load balancing. Kubernetes incorporates load-balancing mechanisms to distribute data traffic for reliable service. Kubernetes automates the deployment, scaling, and management of containerized applications across clusters. It supports declarative resource specification, continuous health monitoring, and resilient recovery from failures.(25) For predictive scaling, machine learning models are used to forecast demand based on historical trends, temporal patterns (e.g., time of day and day of week), and external factors such as weather conditions or planned events.

5. Case Studies

5.1 Data governance and real-time processing integration in Shenzhen, China

Shenzhen is one of China's most advanced smart cities, presenting an example of large-scale big-data processing. The city emphasizes cohesive data governance to integrate municipal agencies, private-sector data providers, and domain-specific services, and monitor traffic, energy, and environmental conditions for public safety. Shenzhen's governance consists of thousands of cross-city data-sharing sites and domain-specific exchange portals, forming a multilayered data-service stack that ingests high-frequency IoT data and delivers analytics. The city's open data service has expanded to 25000 data resources annually, with a growth rate of 18%, enhancing interoperability and analytical capacity.(26)

Real-time data streams are processed by the City Brain (named after the Hangzhou City Brain), which employs distributed edge-cloud processing to minimize latency. Sensors are deployed at more than 32000 intersections to transmit traffic volume, occupancy, and signal-state data at rates of 1–5 times per second. The algorithms implemented in the city's transportation command center optimize intersection delays, reducing average waiting times by up to 12.6% during peak conditions. Edge-layer devices perform essential preprocessing tasks such as signal cleaning, compression, and anomaly filtering, which are critical in ultradense IoT topologies.(26) Environmental and water resource monitoring is centralized in the ecological management system. The city's IoT sensor network includes more than 5400 air-quality sensors and 1200 water-quality monitoring nodes that continuously measure particulate matter, dissolved oxygen, turbidity, and other parameters.(26) Machine-learning-based anomaly detectors are implemented in fog-computing gateways with a detection rate higher than 92%, enabling proactive alerts for emission control and targeted inspections.(27) These capabilities highlight the value of integrating robust data governance with low-latency processing in mission-critical infrastructure management.

Institutional controls over data quality are strong in the city. 74% of public-sector departments experienced improvements in data standardization with their cross-agency metadata scheme. However, 41% expressed concerns regarding the long-term sustainability of the scheme to ensure data accuracy. Despite these limitations, Shenzhen demonstrates how governance frameworks, computational architecture, and sensor network design can collectively enable real-time, city-scale data processing.

5.2 Evolution of smart city policies and their effect on data infrastructure

Since 2012, China's smart city initiative has evolved through three phases, each introducing new technical requirements for IoT infrastructure and data infrastructure.(28)

- Phase I (2011–2013): Focused on digital infrastructure expansion, broadband penetration, fiber-optic coverage, and data center consolidation as foundations for real-time systems.
- Phase II (2014–2016): Emphasis on application-level innovation, with government-supported pilot platforms for transportation analytics, intelligent utilities, and city management. Open data portals expanded significantly, increasing datasets by more than 300%.(29) This period also presented the widespread adoption of time-synchronous IoT protocols and cloud-based big-data platforms such as Spark Streaming and Apache Flink.
- Phase III (2017–2022): Known as cooperative governance, this phase promoted cross-agency data sharing, joint computational platforms, and shared cloud infrastructure. Policy-aligned data exchange architectures reduced latency by 15–22% with multisource IoT streams.

An analysis of 239 documents revealed that structural reforms in governance, data regulation, and technological standards have been critical.[30] Policy effectiveness, measured as the annual average policy effectiveness, tripled to 3.25 in 2012 and became the highest in 2019, coinciding with the introduction of governmental guidelines on data exchange, cloud migration, and cybersecurity standards. The guidelines standardized IoT interoperability and communication architectures and applications across provinces. National policies also presented standards and architectures for real-time data systems, which stimulated engineering and computational practices, and enhanced interoperability, sensor density, and machine learning integration in real-time decision-making.

5.3 Impact of smart city on urban system innovation

The socioeconomic impacts of smart city initiatives in China are closely linked to the deployment of real-time IoT infrastructure and data platforms. A quasi-natural experiment using a multiperiod difference-in-differences model revealed that China's smart city program significantly enhanced municipal innovation capacity. Data from 150 cities revealed that pilot smart cities presented a 7–10% higher increase in the number of patents and technology-based entrepreneurship than other cities. Such results are attributed to the use of real-time sensor data and cloud-based analytics. Informatization intensity (digital infrastructure density and ICT investment) and industrial structure upgrading (growth in sectors reliant on continuous data analytics, such as logistics optimization, intelligent manufacturing, and environmental informatics) also mediate the results. Cities with higher IoT device density (300–400 devices per km^2) exhibited stronger productivity gains in technology sectors.[31]

Evident impacts were observed in pilot smart cities. For example, the Hangzhou City Brain, with integrated video analytics, sensor fusion, and stream processing, reduced emergency response times by 15–23% for diverse city services through improved dispatching and predictive congestion avoidance.[27] Similarly, Suzhou's utility-metering systems reduced the amount of nonrevenue water by 14% using anomaly detection algorithms deployed on edge-fog clusters. Pilot smart cities also achieved cost efficiencies. Their municipal maintenance systems demonstrated predictive maintenance, enabled by the real-time monitoring of noise, temperature, and pressure, and reduced infrastructure downtime by up to 31% and maintenance costs by 9%. These results showed the benefits of real-time data processing and the macroeconomic and operational advantages for the smart city.

6. Challenges and Future Trends

The upcoming technologies offer the opportunity to enhance smart cities' capabilities, as well as a range of new technical and social challenges that will need to be actively considered. Advances in AI and machine learning enable more advanced analytics, such as computer vision to analyze traffic accidents, dangerous pedestrians, and infrastructure flaws; natural language processing to analyze citizen feedback and social media attitudes toward municipal services; and generative AI to simulate scenarios in city planning and optimize infrastructure layouts. The use of federated learning methods facilitates joint model training across multiple cities without centralizing sensitive data while still retaining its privacy elements, and takes advantage of the collective intelligence. Edge AI uses the inference of neural networks on sensor nodes and edge gateways, which lowers latency in time-sensitive uses and reduces bandwidth usage through the local processing of data before transmitting them selectively to the cloud. Digital twins produce virtual models of physical urban infrastructure to support the planning process through simulation and predictive maintenance via physics-informed models, and to evaluate policies by analyzing what-if scenarios before they take place in the real world.

The 5G cellular networks and 6G wireless technologies offer greater connectivity, supporting massive IoT implementation through network slicing, which allows customized virtual networks to optimize specific application needs such as ultrareliable low-latency communication and massive machine-type communication. Quantum computing has potential for addressing complex optimization challenges, including traffic routing, resource allocation, and energy grid management and complementing classical computing in tasks that demand advanced computational capabilities for urban management. Such possible applications might highlight the transformative role of quantum technologies in solving intricate environmental and infrastructural problems.

Major obstacles threaten the sustainability of smart cities and the distribution of benefits, and these should be addressed with multidisciplinary solutions across technical, policy, and social spheres. Any security weaknesses in IoT devices, such as weak authentication, unpatched software, and flawed communication systems, will provide attack points for malicious individuals targeting vital urban infrastructure. The cases of malware infections, large-scale IoT botnets, and similar incidents indicate that poorly secured devices can be used to launch distributed denial-of-service attacks. The privacy associated with pervasive surveillance using camera networks, location tracking, and behavior analytics would require well-structured governance frameworks that spell out data collection restrictions, retention, purpose restrictions, and individual rights, such as transparency and consent guidelines. Differential privacy methods that add noise to aggregate statistics, calibrated appropriately, enable useful analytics without the need for individual re-identification. However, their complexity and the balance between utility and privacy must be carefully configured. Machine learning models trained on historically biased data are prone to perpetuating or exaggerating socio-equity-based health inequalities through automated decision-making in resource allocation, service prioritization, and enforcement actions.

Therefore, a fair-awareness algorithm design and disparate impact monitoring across and within demographic groups are needed. Scalability issues arise when pilot projects become city-wide deployments and experience significant increases in device populations, data volumes, and user bases, requiring architectural designs that support horizontal scaling and overcome bottlenecks through distributed processing and storage. Computation-intensive analytics and communication infrastructure energy use create sustainability problems, which in turn drive energy-efficient algorithm design, integration with renewable energy, and workload scheduling aligned with generation profiles. Sustainability demands a proposition that is worth pursuing, given operational expenses and the need to break technology to ensure continuity; this is made difficult by long infrastructure lifecycles and the rapid pace of technological change, which can make investments seem outdated. Full mutualism between smart city networks across different jurisdictions would enable positive data sharing and allow the unified management of the entire region, but requires standardization that remains cost-effective and consistent.

7. Conclusions

We explored the role of real-time data processing in the development of smart city infrastructures. The integration of sensor networks, ranging from environmental probes to LiDAR, requires sophisticated architectures that process and analyze massive, asynchronous data streams. By employing edge analytics for data filtering and compression and time-series databases, smart cities significantly reduce bandwidth usage and latency while maintaining analytical utility.

The transition to real-time processing enables adaptive systems, such as traffic management networks that adjust signal timing dynamically on the basis of traffic flow rates. Efficient resource allocation can be achieved on the basis of reinforcement learning strategies and automated container orchestration, allowing systems to predict demand on the basis of historical trends and external factors. Shenzhen's governance and infrastructure demonstrate the benefits of these technologies. The coordination between robust data governance and technical infrastructure, such as the deployment of more than 32000 traffic sensors, has presented significant improvements in public-service efficiency and environmental monitoring. Pilot smart cities in China have shown faster growth in patent output and technology entrepreneurship than other cities, driven by the productivity gains from the real-time IoT infrastructure.

Despite these advancements, for the sustainability of smart cities, hurdles regarding security, privacy, and energy consumption must be overcome. The proliferation of IoT devices increases malicious attacks, necessitating strict governance, pervasive surveillance safeguards, and privacy-preserving techniques such as differential privacy. The integration of emerging technologies such as generative AI for scenario simulation, digital twins for predictive maintenance, and 6G networks for massive machine-type communication is expected to enhance urban management. To fully realize these benefits, energy-efficient algorithms and standardized cross-agency policies must be prioritized to ensure scalable and secure urban ecosystems.

References

1 C. Karri, J. J. M. Machado, J. M. R. S. Tavares, D. K. Jain, S. Dannana, S. K. Gottapu, and A. H. Gandomi: Comput. Mater. Con. **81** (2024) 3617. https://doi.org/10.32604/cmc.2024.058461
2 United Nations: https://www.un.org/uk/desa/68-world-population-projected-live-urban-areas-2050-says-un (accessed December 2025).
3 G. Tricomi, M. Giacobbe, I. Ficili, N. Peditto, and A. Puliafito: Sensors **24** (2024) 3108. https://doi.org/10.3390/s24103108
4 Y. B. Zikria, H. Yu, M. K. Afzal, M. H. Rehmani, and O. Hahm: Future Gener. Comput. Syst. **88** (2018) 3699. https://doi.org/10.1016/j.future.2018.07.058
5 B. C. Ko, M.-C. Chang, J. T. Lee, J. W. Chong, and J. S. Choi: ETRI J. **47** (2025) 7933. https://doi.org/10.4218/etr2.70073?urlappend=%3Futm_source%3Dresearchgate.net%26utm_medium%3Darticle
6 S.-H. Sim and J.-J. Lee: Sensors **21** (2021) 7064. https://doi.org/10.3390/s21217064
7 M. Talebkhah, A. Sali, M. Marjani, M. Gordan, S. J. Hashim, and F. Z. Rokhani: IEEE Access **9** (2021) 55465. https://doi.org/10.1109/ACCESS.2021.3070905

8 X. Li, H. Liu, W. Wang, Y. Zheng, H. Lv, and Z. Lv: Future Gener. Comput. Syst. **128** (2022) 167. https://doi.org/10.1016/j.future.2021.10.006
9 S. Khan, S. Nazir, I. García-Magariño, and A. Hussain: Comput. Electr. Eng. **89** (2021) 106906. https://doi.org/10.1016/j.compeleceng.2020.106906
10 B. N. Silva, M. Khan, C. Jung, J. Seo, D. Muhammad, J. Han, Y. Yoon, and K. Han. Sensors **18** (2018) 2994. https://doi.org/10.3390/s18092994
11 A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi: IEEE Internet Things J. **1** (2014) 22. https://ieeexplore.ieee.org/document/6740844
12 I. Zyrianoff, A. Heideker, D. Silva, J. Kleinschmidt, J.-P. Soininen, T. S. Cinotti, and C. Kamienski: Sensors **20** (2019) 84. https://doi.org/10.3390/s20010084
13 H. Mrabet, S. Belguith, A. Alhomoud, and A. JemaiL Sensors **20** (2020) 3625. https://doi.org/10.3390/s20133625
14 B. Tekinerdogan, Ö. Köksal, and T. Çelik: Appl. Sci. **13** (2023) 4173. https://doi.org/10.3390/app13074173
15 Z. Khalid, N. Fisal, and Mohd. Rozaini: Sensors **14** (2014) 24046. https://doi.org/10.3390/s141224046
16 S. Henning and W. Hasselbring: J. Syst. Softw. **208** (2024) 111879. https://doi.org/10.1016/j.jss.2023.111879
17 A. AlOrbani and M. Bauer: Proc. 2021 IEEE Int. Smart Cities Conf. (ISC2, 2021) 1. https://doi.org/10.1109/isc253183.2021.9562941
18 A. Rejeb, K. Rejeb, H. Treiblmaier, A. Appolloni, S. Alghamdi, Y. Alhasawi, and M. Iranmanesh: IoT **22** (2023) 100721. https://doi.org/10.1016/j.iot.2023.100721
19 P. Su, Y. Chen, and M. Lu: J. Supercomput. **78** (2021) 3676. https://doi.org/10.1007/s11227-021-03972-5
20 S. Sampaio, P. Sousa, C. Martins, A. Ferreira, L. Antunes, and R. Cruz-Correia: Appl. Sci. **13** (2023) 3830. https://doi.org/10.3390/app13063830
21 J. Verwiebe, P. M. Grulich, J. Traub, and V. Markl: VLDB J. **32** (2023) 985. https://doi.org/10.1007/s00778-022-00778-6
22 S. Ugwuanyi, G. Paul, and J. Irvine: Electronics **10** (2021) 2224. https://doi.org/10.3390/electronics10182224
23 S. Azzabi, Z. Alfughi, and A. Ouda; Computers **13** (2024) 183. https://doi.org/10.3390/computers13070183
24 K. Ahmad, M. Maabreh, M. Ghaly, K. Khan, J. Qadir, and A. Al-Fuqaha: Comput. Sci. Rev. **43** (2022) 100452. https://doi.org/10.1016/j.cosrev.2021.100452
25 C. Carrión: ACM Comput. Surv. **55** (2022) 1. https://doi.org/10.1145/3539606
26 S. Xie, N. Luo, and M. Yarime: PDP **7** (2022) 66. https://doi.org/10.1080/25741292.2023.2297445
27 D. D. Sanchez-Gallegos, D. E. Carrizales-Espinoza, C. Torres-Charles, and J. Carretero: Smart Cities **8** (2025) 173. https://doi.org/10.3390/smartcities8050173
28 E. J. Sacoto-Cabrera, A. Perez-Torres, L. Tello-Oquendo, and M. Cerrada: Smart Cities **8** (2025) 175. https://doi.org/10.3390/smartcities8050175
29 A. Rahman, M. A. Rouf, N. Sultana, and M. S. Akter: JSDP **1** (2025) 144. https://doi.org/10.63125/6y0qwr92
30 C. Yue, H. Li, H. Mao, and A. Yue: Buildings **15** (2024) 7. https://doi.org/10.3390/buildings15010007
31 S. Chen, X. Zhang, W. Wang, Y. Liang, W. He, and Z. Tan: Sustainability **16** (2024), 8007. https://doi.org/10.3390/su16188007