Challenges in Edge Computing II

Chapter 7

Topic: Internet of Things | Retail | Autonomous Vehicles

Outline

- Internet of Things
  - Mobile gaming & IIoT
  - MEC Applications
  - Security and Data Sharing
- Retail
  - Smart Retail Applications
  - SmSH Architecture
  - RFID / NFC / Bluetooth / LPWAN
- Autonomous Vehicles
  - OpenVDAP & VECFrame
  - Federated Learning & Computation Offloading
  - SafeCross & C-V2X CCAD
- Summary & Practice

Internet of Things

Bringing computational resources closer to data sources cuts latency and boosts data-processing efficiency — vital for real-time IoT.

Mobile gaming

Cloud IoT suffers high latency & unreliable links — edge architectures cut latency and lift quality of experience for mobile gaming.

Industrial IoT

Decentralize processing to cut latency & bandwidth; full/partial offloading + edge AI handle complex computation.

Deep learning on edge

Deploy DL models partly on edge servers with a scheduling strategy that maximizes tasks under quality-of-service limits.

Secure data sharing

Blockchain is proposed for secure IoT data sharing; future work targets 5G network slicing, better load balancing & robust security.

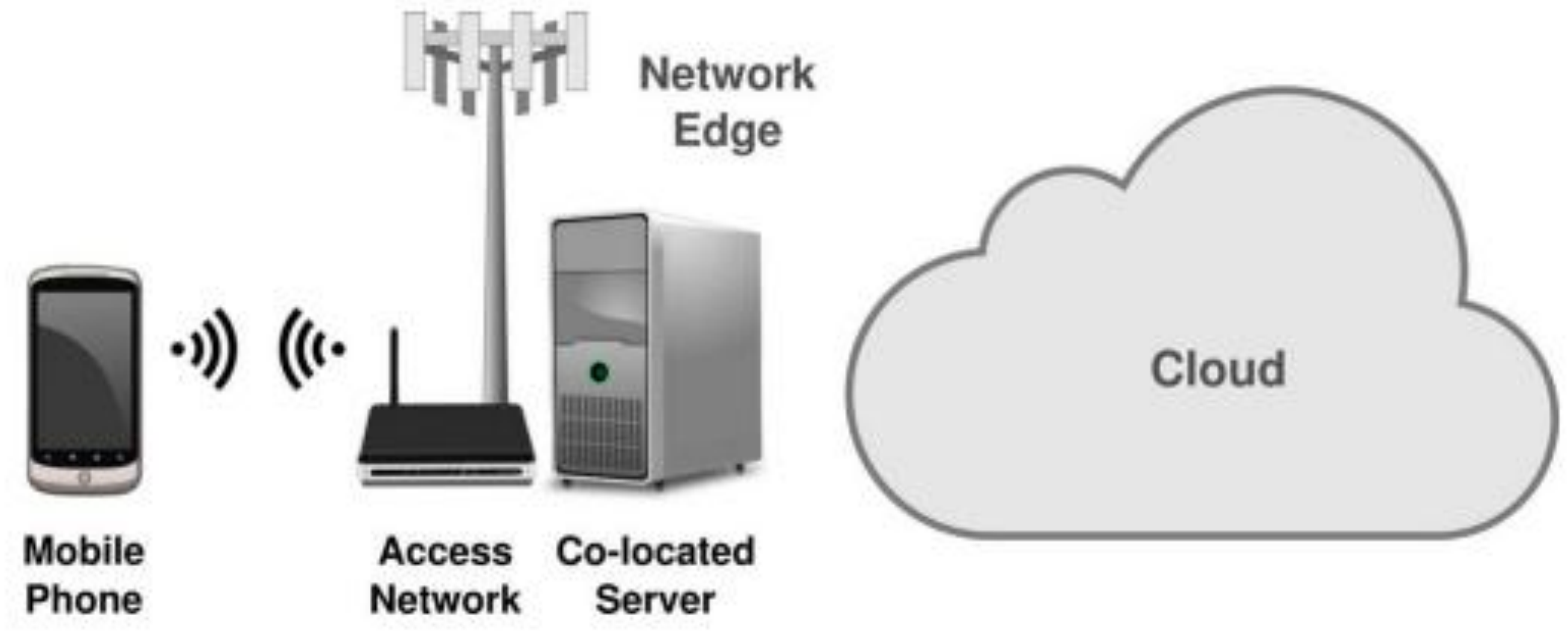
Gopika Premsankar, Mario Di Francesco, and Tarik Taleb. “Edge computing for the Internet of Things: A case study”. In: IEEE Internet of Things Journal 5. 2 (2018), pp. 1275–1284.

Tie Qiu et al. “Edge computing in Industrial Internet of Things: Architecture, advances and challenges”. In: IEEE Communications Surveys & Tutorials 22. 4 (2020), pp. 2462–2488.

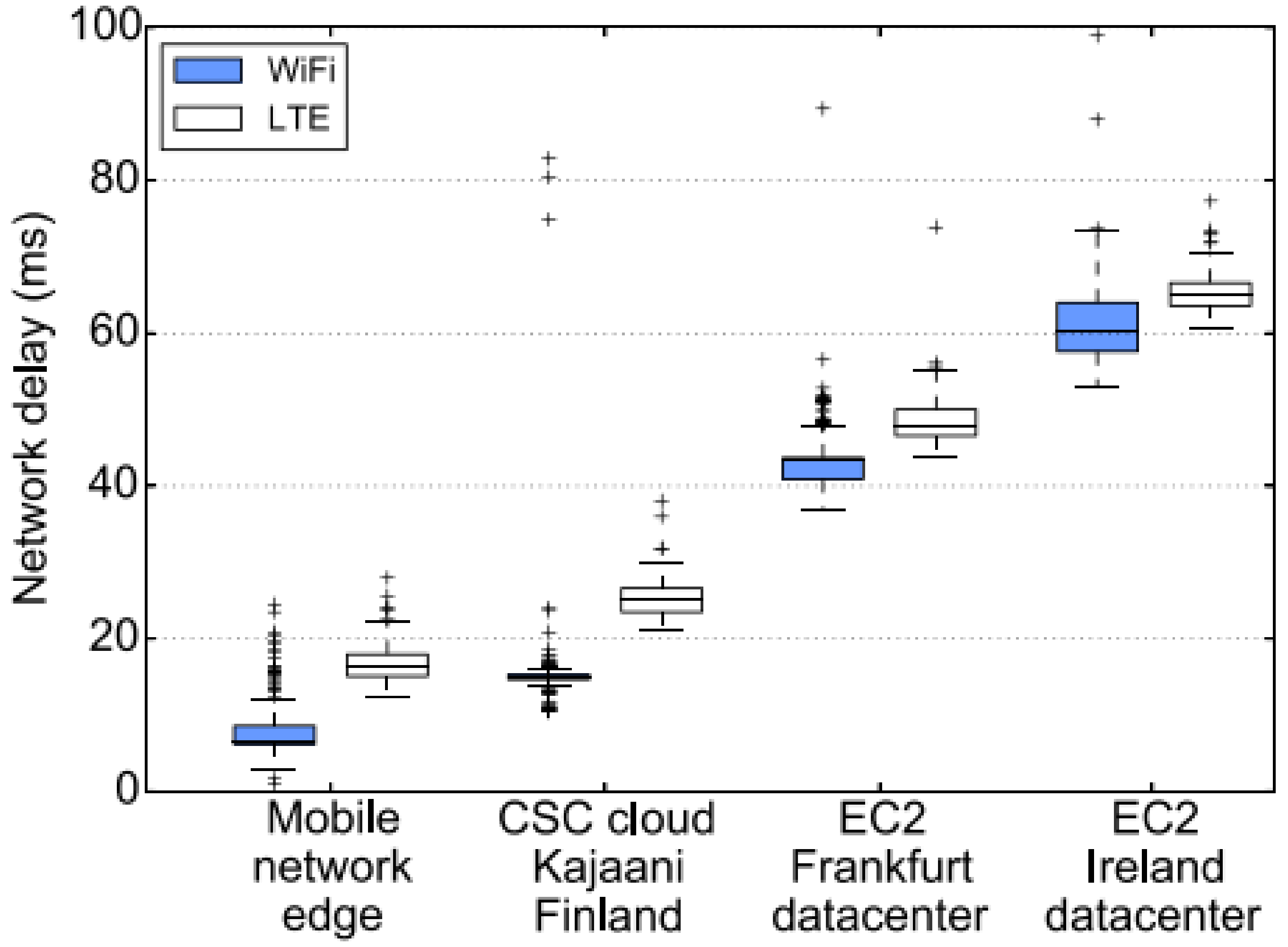
J Lin et al. “An edge computing based public vehicle system for smart transportation”. In: IEEE Transactions on Vehicular Technology 69. 11 (2020), pp. 12635–12651. DOI: 10.1109/TVT.2020.3028497.

Internet of Things

Mobile gaming



Network edge testbed setup



Network delay across server deployments

➢ Key characteristics

Application Mobile gaming / AR-VR-style interactive workloads

Why edge Cloud latency hurts responsiveness and user experience

Approach Offload rendering and processing to a nearby edge server

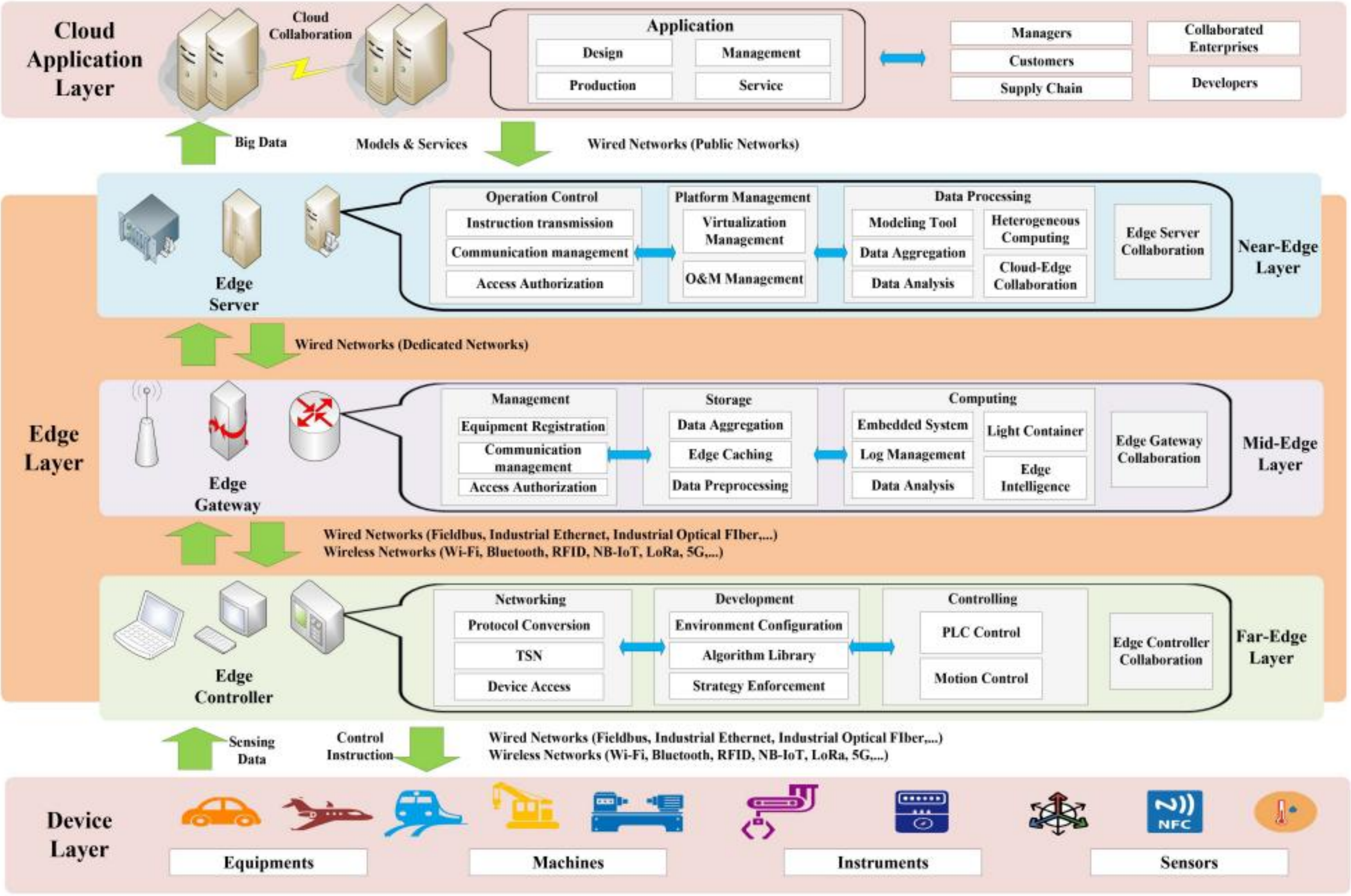
Benefit Edge deployment provides much lower network delay than distant cloud servers

Network delay <20 ms at mobile edge vs. ~40–70 ms at public cloud

Gopika Premsankar, Mario Di Francesco, and Tarik Taleb. “Edge computing for the Internet of Things: A case study”. In: IEEE Internet of Things Journal 5. 2 (2018), pp. 1275–1284.

Internet of Things

Industrial IoT (IIoT)



Edge Computing in IIoT

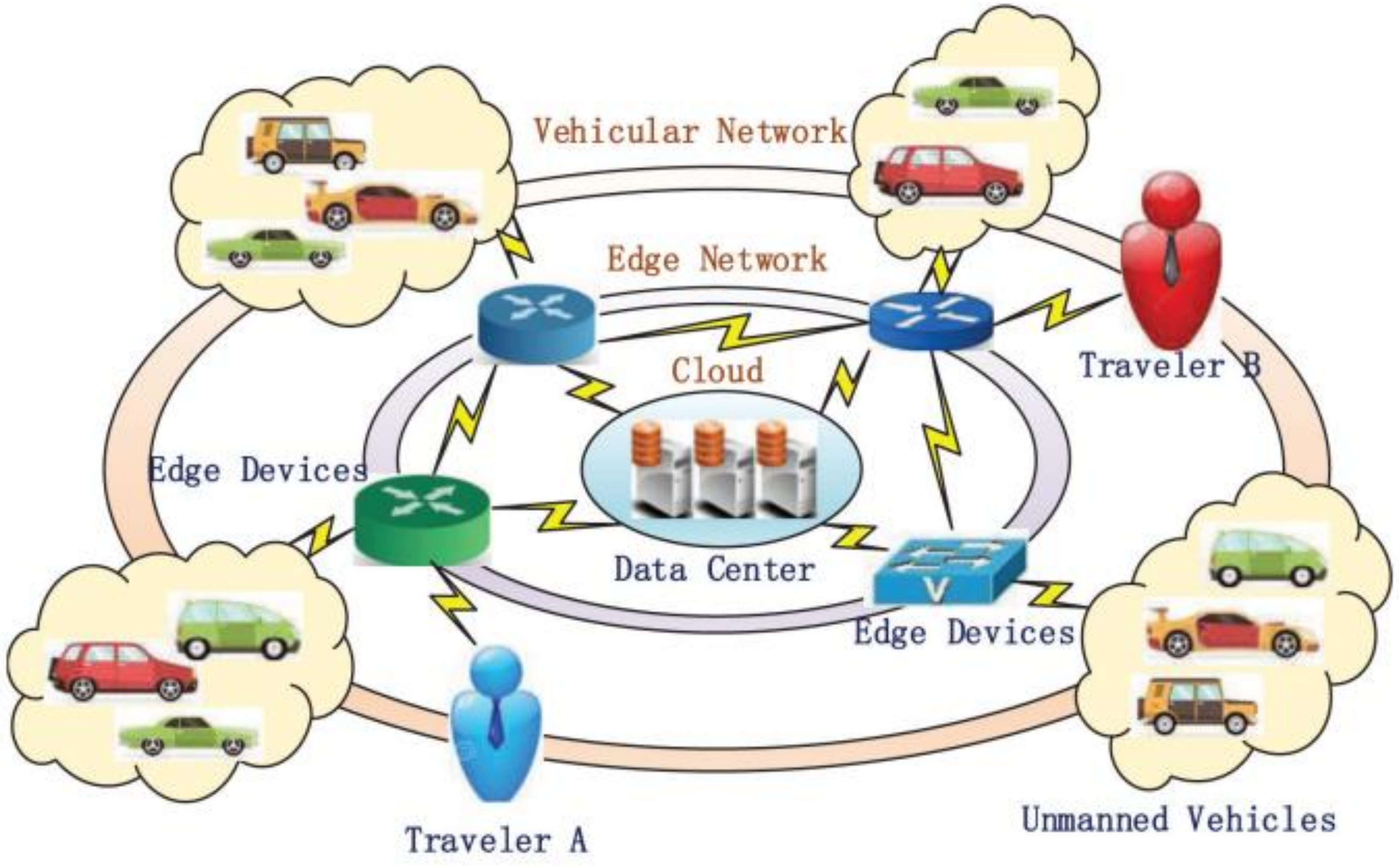
Feature	IoT	IIoT
Objective	Consumption-oriented	Production-oriented
Application scenarios	Smart home, health, localization	Smart factory, logistics, remote
Platform foundation	Self-building	Industrial facilities
Sensors scale	Small to medium	Large to very large
Precision requirement	Low to medium	High to very high
Mobility	Strong	Weak
Data volume	Medium to high	High to very high
Latency tolerance	High	Very low

Qualitative comparison of IoT and IIoT

Tie Qiu et al. “Edge computing in Industrial Internet of Things: Architecture, advances and challenges”. In: IEEE Communications Surveys & Tutorials 22. 4 (2020), pp. 2462–2488.

Internet of Things

Public vehicle system for smart transportation



System model for smart transportation

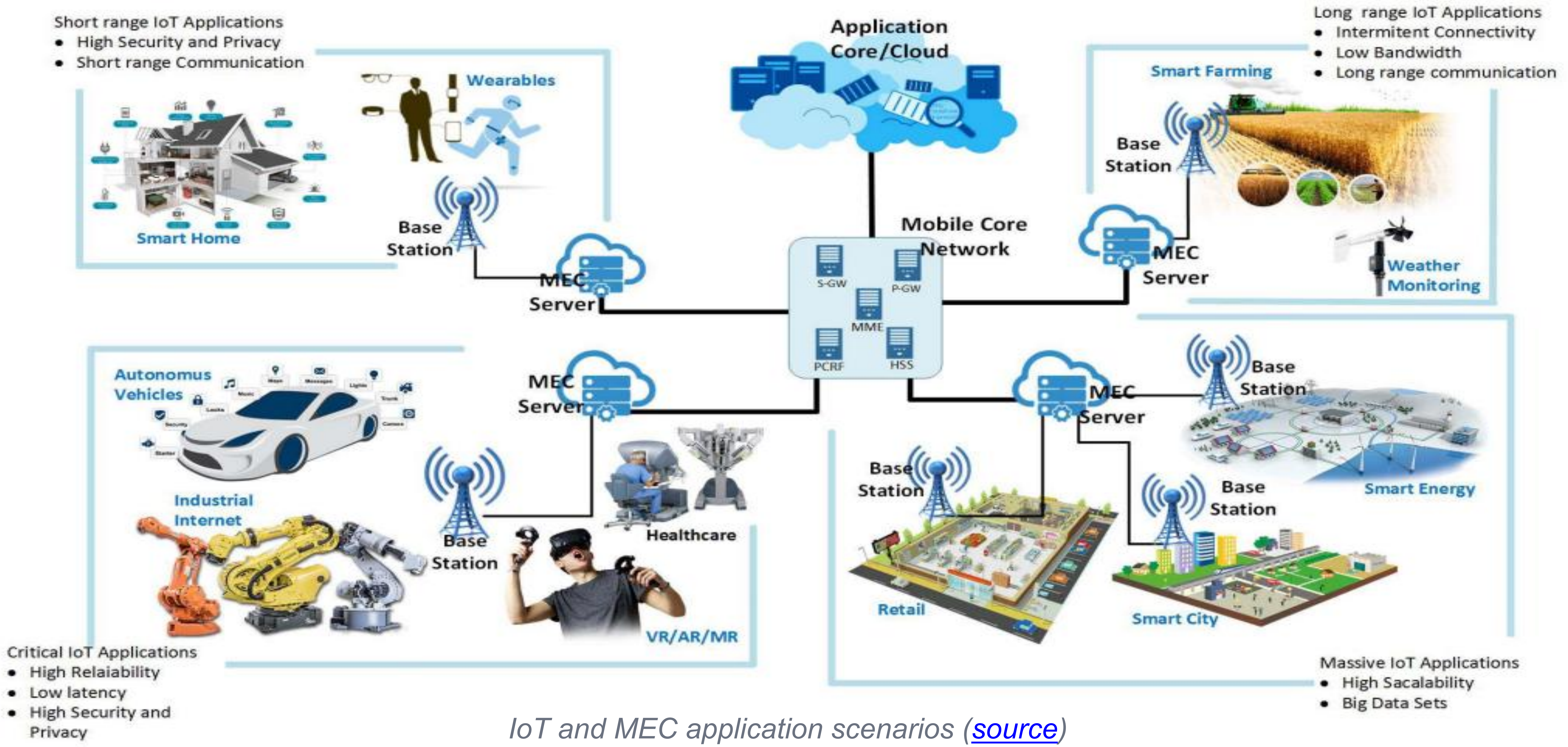
➢ Key characteristics

- **Rapid edge-benefit estimation** helps determine whether moving an application to the edge is actually worthwhile.
- **Castnet framework** estimates edge-computing gains without requiring a full real-world edge deployment.
- Evaluates the trade-off between **application responsiveness, network/backhaul savings, and edge infrastructure cost.**
- Targets **MEC deployment planning and what-if analysis** before committing infrastructure resources.

J Lin et al. “An edge computing based public vehicle system for smart transportation”. In: IEEE Transactions on Vehicular Technology 69. 11 (2020), pp. 12635–12651. DOI: 10.1109/TVT.2020.3028497.

Internet of Things

Application scenario for IoT and multi access edge computing (MEC)



IoT and MEC application scenarios (source)

Pawani Porambage et al. "Survey on multi-access edge computing for Internet of Things realization". In: IEEE Communications Surveys & Tutorials 20. 4 (2018), pp. 2961–2991.

Internet of Things: Challenges

Offloading & load balancing
Distributed IoT devices need advanced load-balancing algorithms.

Limited edge resources
Constrained compute at the edge caps what runs locally.

Security & privacy
Robust protection — blockchain for secure data sharing.

5G integration
Network slicing & 5G integration still being optimized.

Retail

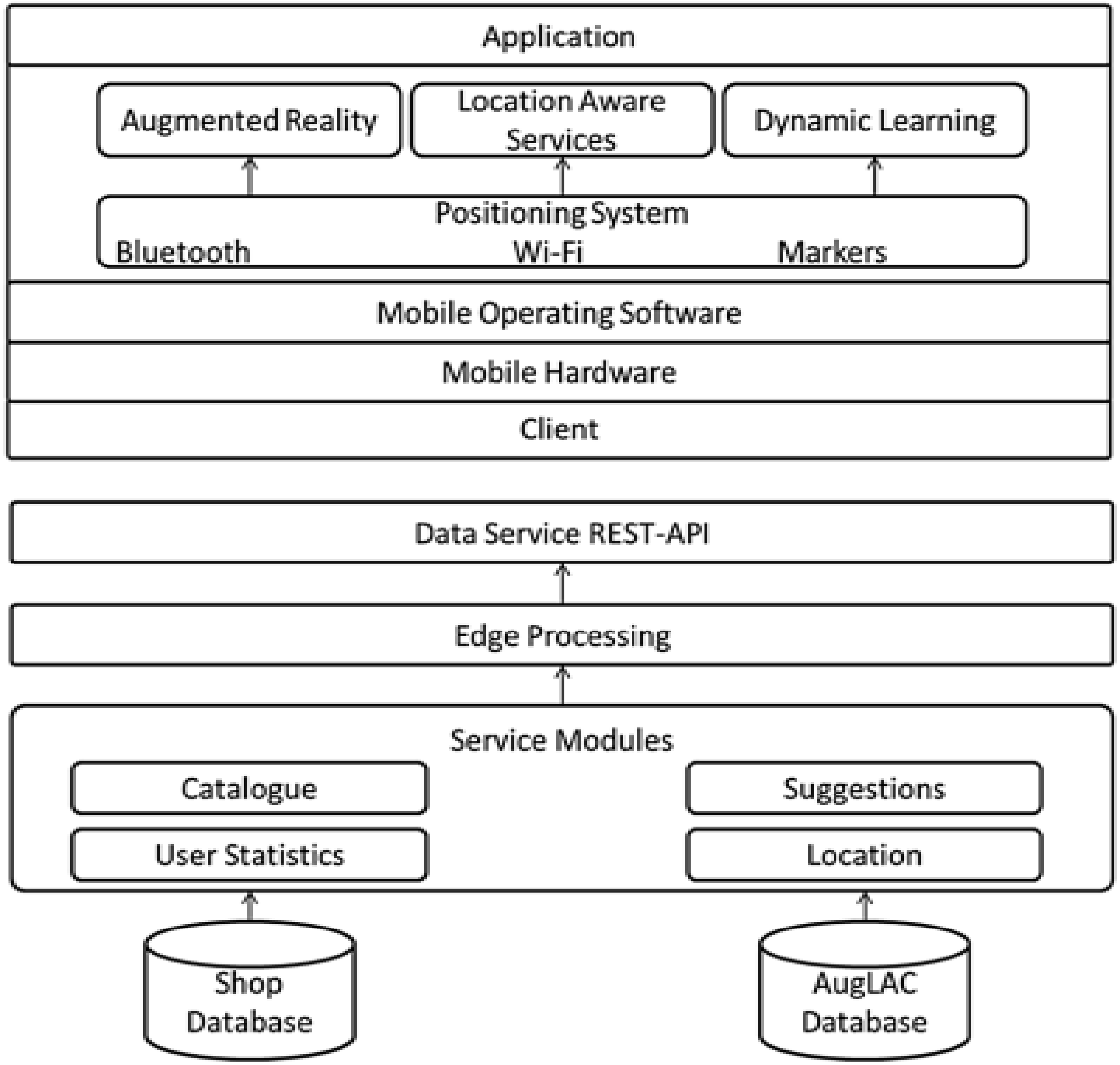
High-level view of an IoT-based smart retail

➢ **Six edge-enabled retail functions**

- **Smart advertisement** Targeted, real-time promotions in-store.
- **Smart retail store** Track customer traffic & buying patterns locally.
- **Smart inventory** Weight sensors + cameras + RFID auto-signal restock.
- **Smart checkout** Faster, frictionless payment at the edge.
- **Smart logistics** Local processing streamlines supply & delivery.
- **Smart business analysis** On-site analytics for immediate decisions.

Charith Perera et al. "Fog computing for sustainable smart cities: A survey". In: ACM Computing Surveys (CSUR) 50. 3 (2017), pp. 1–43.

Retail



SmSH architecture

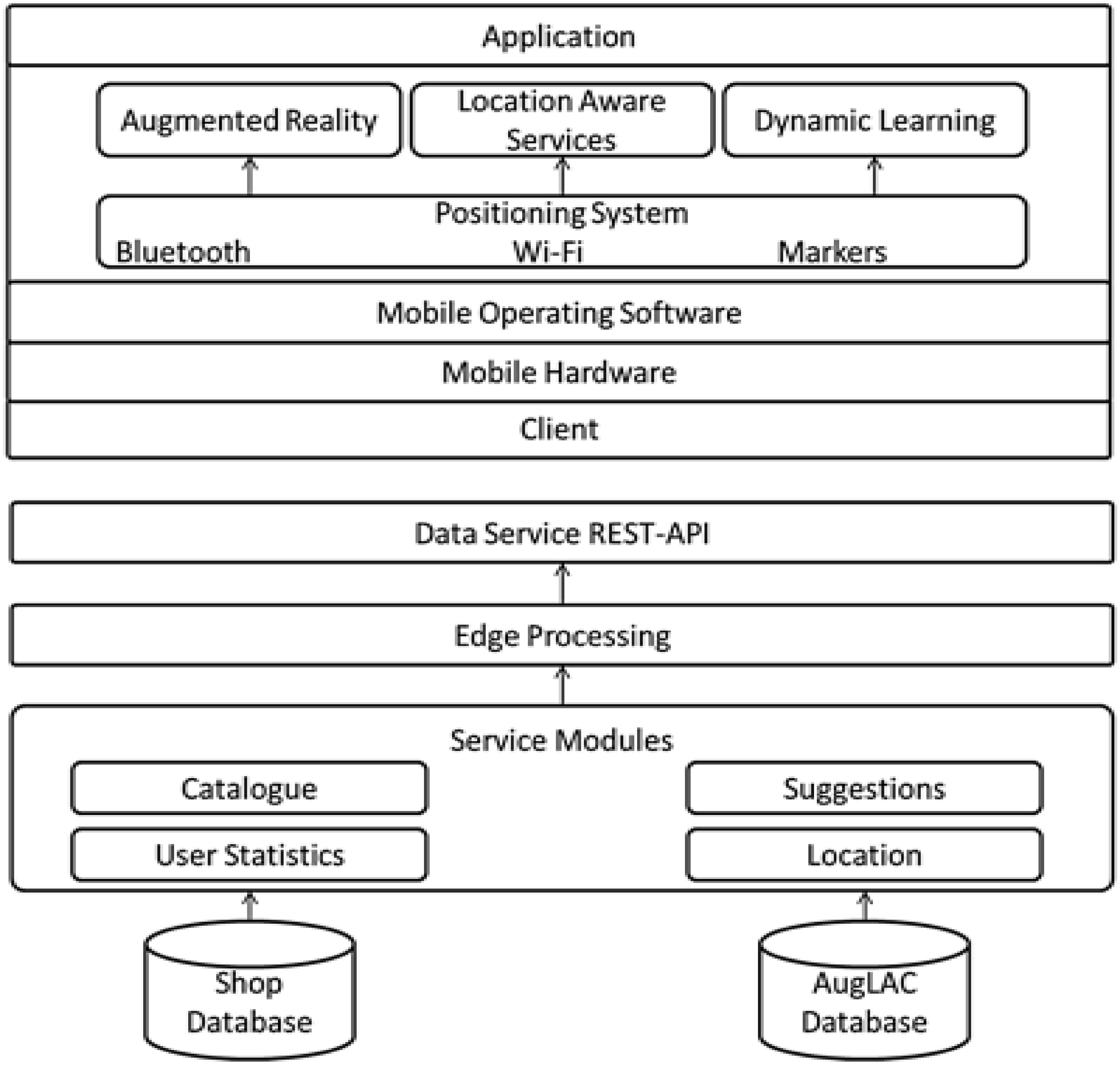
Smart shop (SmSH). A Smart Shop architecture integrating IoT devices + edge computing for seamless device communication, real-time processing and personalized customer interaction.

Connectivity technologies

- **RFID** Tag-based item identification & tracking.
- **NFC** Near-field taps for checkout & pairing.
- **Bluetooth** Short-range device-to-device links.
- **LPWAN** Low-power wide-area coverage for sensors.

M Saravanan and N C Srinidhi Srivatsan. "Intelligent communication between IoT devices on edges in retail sector". In: Advances in Information and Communication Networks: Proceedings of the 2018 Future of Information and Communication Conference (FICC), Vol. 2. Springer, 2019, pp. 546–562.

Retail



SmSH architecture (source)

Smart shop (SmSH). A Smart Shop architecture integrating IoT devices + edge computing for seamless device communication, real-time processing and personalized customer interaction.

➢ Implementation phases

1. **Customer identification** Recognize the shopper on entry.
2. **Product selection** Track picks & assist in real time.
3. **Payment** Fast, secure edge-handled checkout.

Reduces latency, enhances security, and extends easily to other business domains.

M Saravanan and N C Srinidhi Srivatsan. “Intelligent communication between IoT devices on edges in retail sector”. In: Advances in Information and Communication Networks: Proceedings of the 2018 Future of Information and Communication Conference (FICC), Vol. 2. Springer, 2019, pp. 546–562.

Retail: Challenges

Low latency

Real-time response to customer behavior in-store.

Data management

Analyzing & using real-time data at store level.

Security

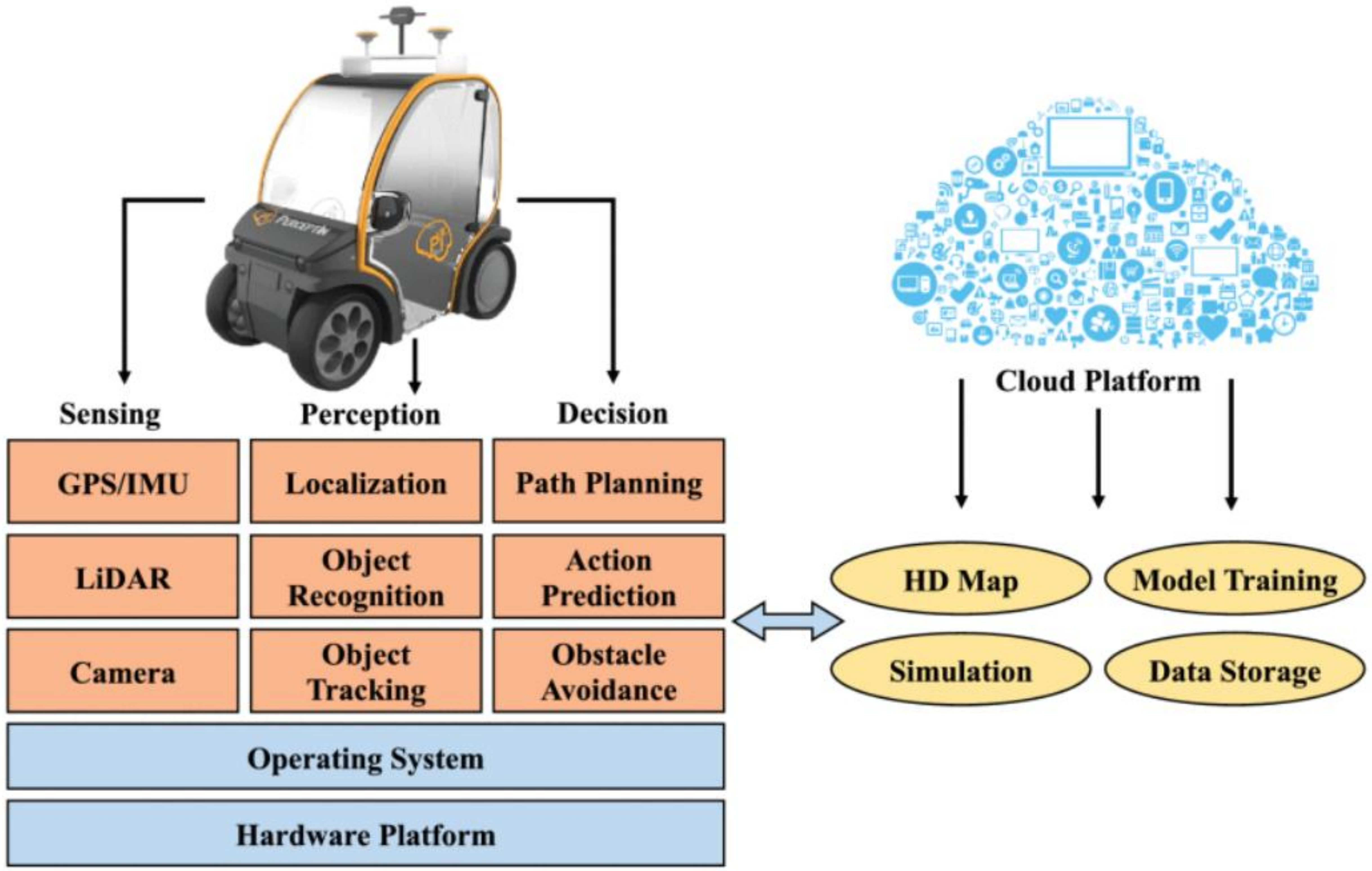
Secure device communication & payment handling.

Competition pressure

Brick-and-mortar must match online personalization.

Autonomous Vehicles

The Connected & Autonomous Vehicle Stack (CAVs)



High-level view of edge computing for autonomous vehicles

Sensing
GPS/IMU, LiDAR, cameras capture the environment.

Perception
Localization, object recognition & tracking.

Decision
Path planning, action prediction, obstacle avoidance.

Cloud platform
HD maps, simulation, model training, data storage.

Shaoshan Liu et al. “Edge computing for autonomous driving: Opportunities and challenges”. In: Proceedings of the IEEE 107. 8 (2019), pp. 1697–1716.

Autonomous Vehicles

Edge frameworks for autonomous driving

OpenVDAP

Full-stack platform: onboard heterogeneous unit + secure EdgeOSv + edge-aware libvdap; dynamically offloads tasks.

VECFrame

Cooperative object detection & data fusion at the edge via modular containers — boosts throughput 40–350% vs. non-edge.

CLONE

Federated learning + LSTM: vehicles train locally, share parameters to an edge server. Predicts EV battery failure.

Offloading

ILP offline + heuristic online offloading to roadside cloud — cuts average latency 34%, lifts localization accuracy up to 7.6×.

SafeCross

Left-turn blind-area warnings via video preprocessing, classification, few-shot learning & model switching.

C-V2X CCAD

Cellular V2X collaborative driving — multi-angle perception & edge processing improve lane-keeping and safety.

Qingyang Zhang et al. “OpenVDAP: An open vehicular data analytics platform for CAVs”. In: 2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS). IEEE, 2018, pp. 1310–1320.

S Tang et al. “VECFrame: A vehicular edge computing framework for connected autonomous vehicles”. In: 2021 IEEE International Conference on Edge Computing (EDGE). 2021, pp. 68–77. DOI: 10.1109/EDGE53862.2021.00019.

Sidi Lu, Yongtao Yao, and Weisong Shi. “Collaborative learning on the edges: A case study on connected vehicles”. In: 2nd USENIX Workshop on Hot Topics in Edge Computing (HotEdge 19), 2019.

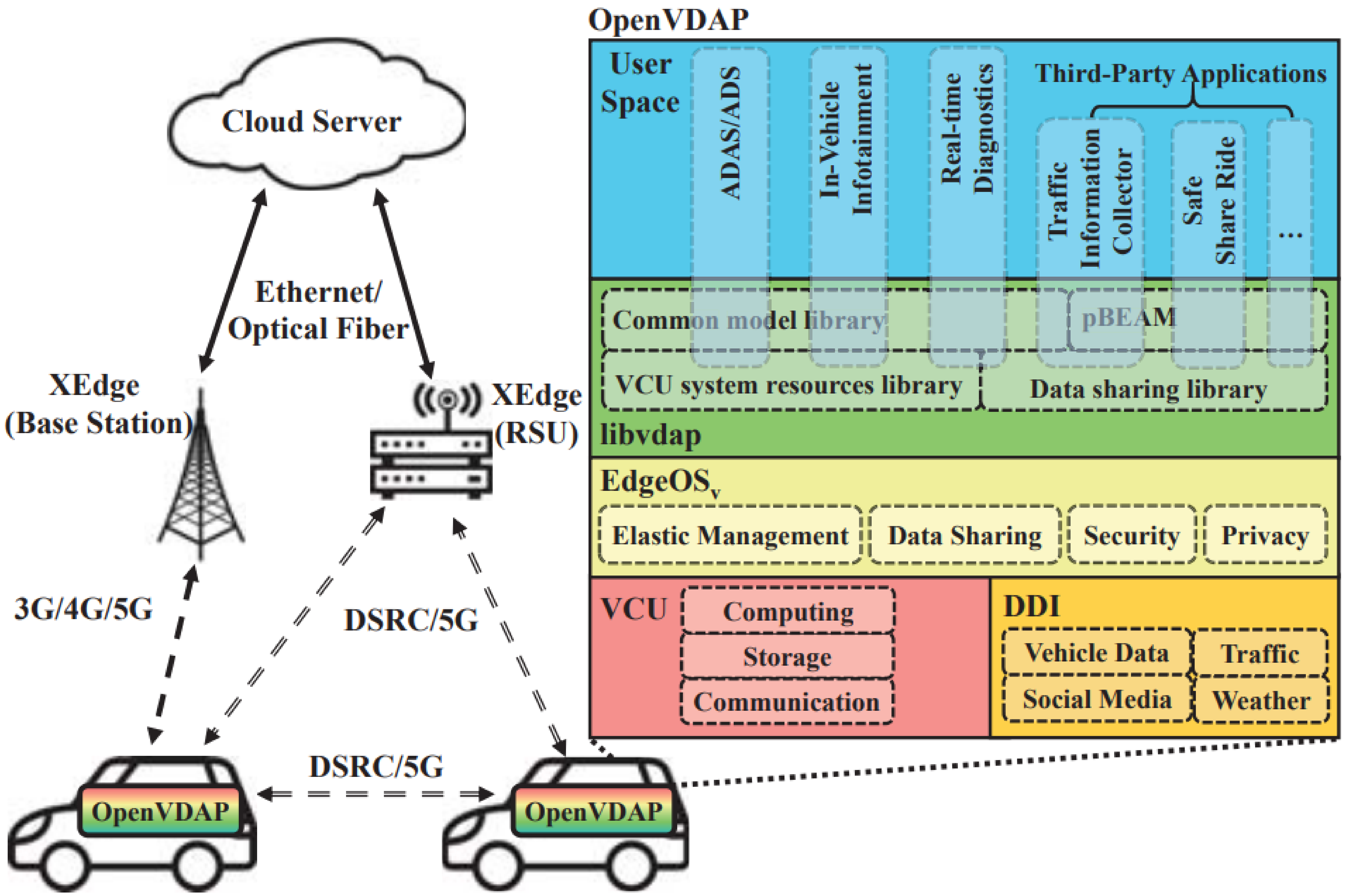
Mingyue Cui et al. “Offloading autonomous driving services via edge computing”. In: IEEE Internet of Things Journal 7. 10 (2020), pp. 10535–10547.

B Wu et al. “To turn or not to turn, safecross is the answer”. In: 2022 IEEE 42nd International Conference on Distributed Computing Systems (ICDCS). 2022, pp. 414–424. DOI: 10.1109/ ICDCS54860.2022.00047.

Y He et al. “Towards C-V2X enabled collaborative autonomous driving”. In: IEEE Transactions on Vehicular Technology 72. 12 (2023), pp. 15450–15462. DOI: 10.1109/ TVT.2023.3299844.

Autonomous Vehicles

OpenVDAP



Overview of OpenDAP (source)

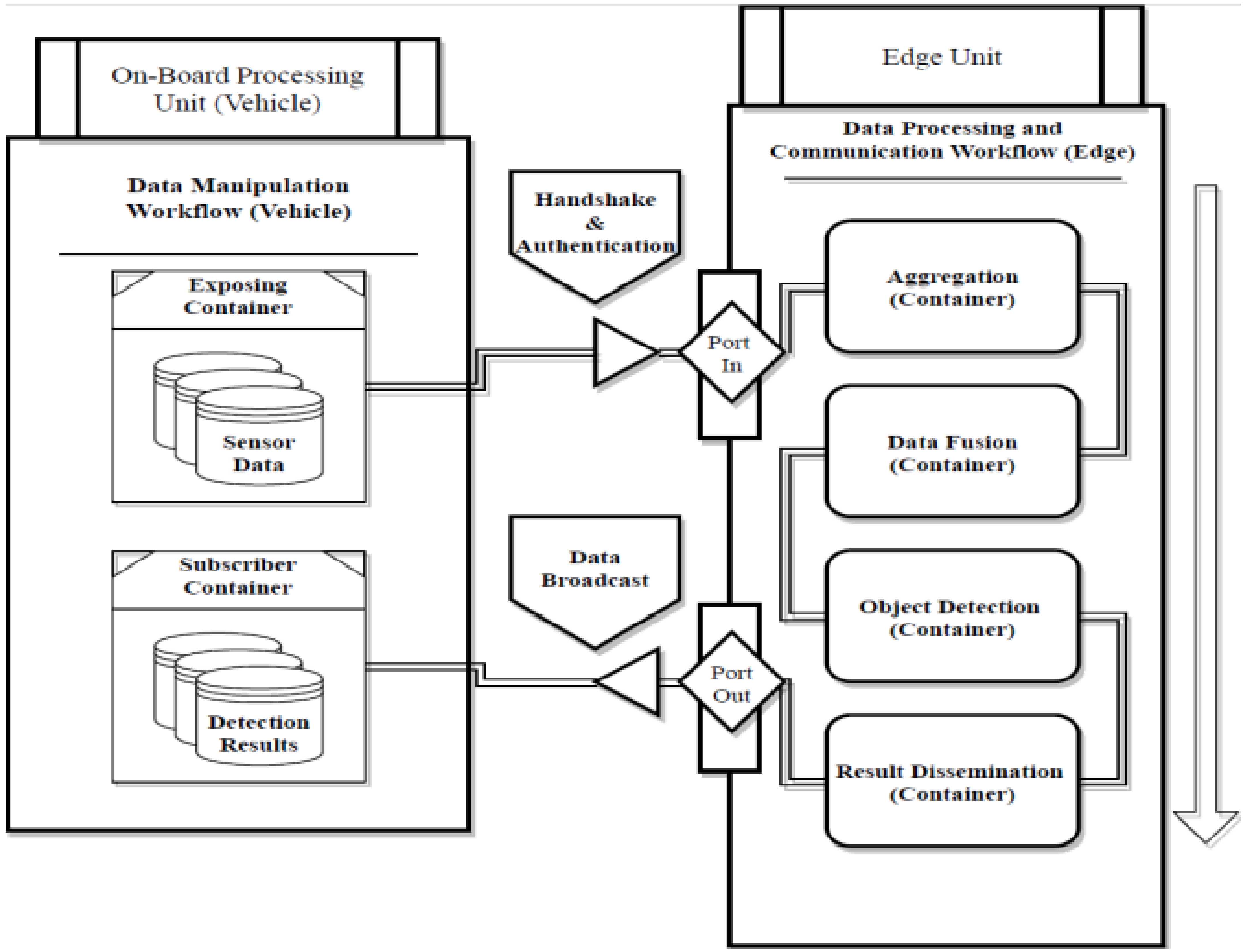
➢ Key characteristics

- Supports **dynamic workload offloading and scheduling** based on latency, bandwidth, and available computing resources.
- Uses **a heterogeneous Vehicle Computing Unit (VCU)** with CPU, GPU, FPGA, ASIC, storage, and multiple communication interfaces.
- Introduces **EdgeOSv** for elastic resource management, isolation, security, privacy, and data sharing.
- Provides **libvdap**, an edge-aware application library with common AI models and uniform APIs for developers.
- Designed as an **open-source platform** with APIs and real vehicle data for developing and testing CAV applications.

Qingyang Zhang et al. “OpenVDAP: An open vehicular data analytics platform for CAVs”. In: 2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS). IEEE, 2018, pp. 1310–1320.

Autonomous Vehicles

VECFrame



VECFrame container-based workflows (source)

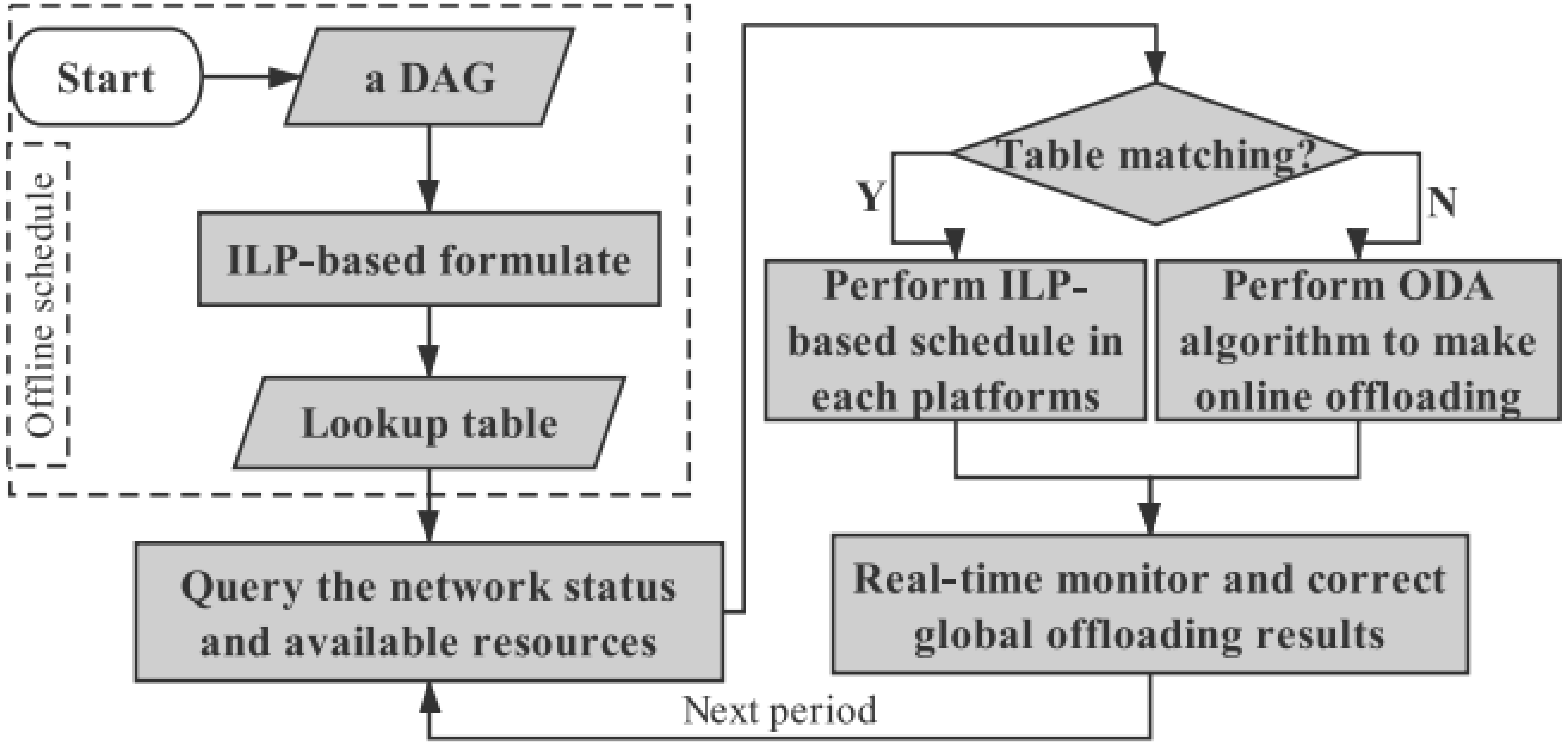
➢ Key characteristics

- **Vehicular edge computing framework** for CAVs that combines vehicle, edge-node, and infrastructure resources for cooperative perception.
- Designed around three principles: **data-friendly, adaptable, and scalable.**
- Employs a **publish–subscribe communication model** using MQTT, with Wget used for reliable raw-data transfer.
- Supports **multi-vehicle data aggregation, sensor fusion, object detection, and result dissemination** at the edge.
- Experimental results show **41.7% higher MQTT throughput and 350% higher Wget throughput** when using edge nodes.

S Tang et al. "VECFrame: A vehicular edge computing framework for connected autonomous vehicles". In: 2021 IEEE International Conference on Edge Computing (EDGE). 2021, pp. 68–77. DOI: 10.1109/EDGE53862.2021.00019.

Autonomous Vehicles

Offloading autonomous driving via edge computing



Overview of offloading autonomous driving services via edge computing (source)

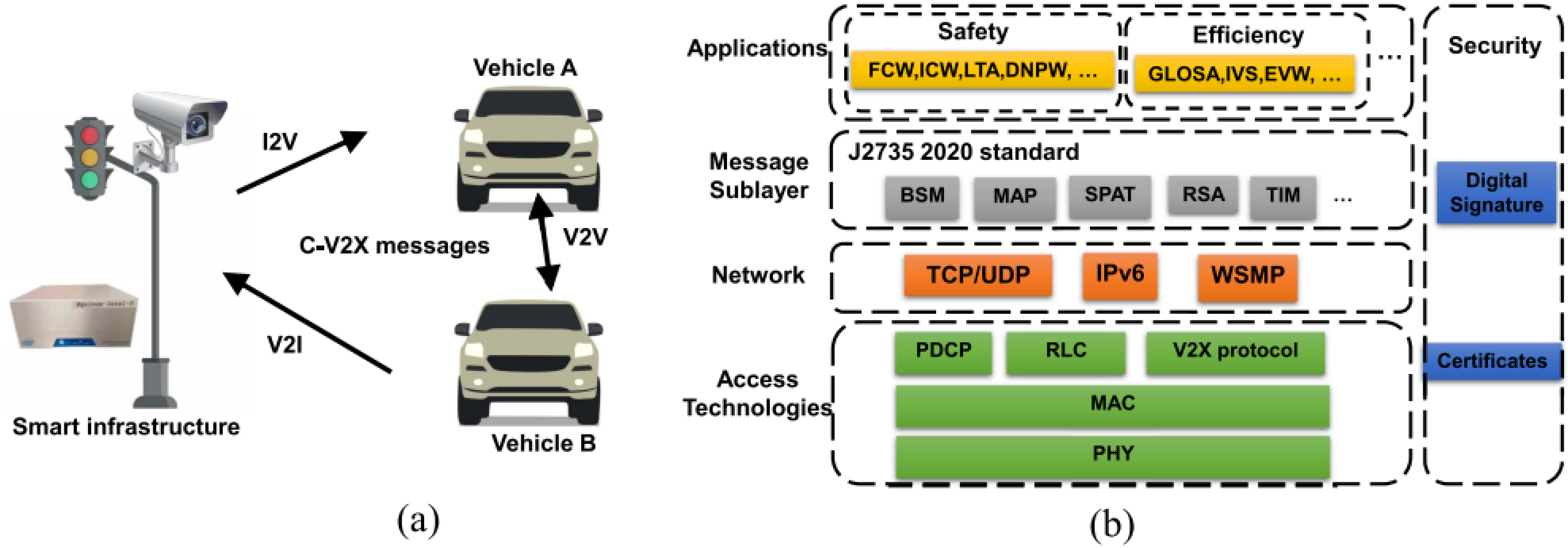
➢ Key characteristics

- **Three-layer OBU–Edge–Cloud architecture** for offloading autonomous-driving computation tasks.
- Models autonomous-driving services as **DAG-based dependent tasks** with execution deadlines and communication dependencies.
- Uses **offline ILP optimization** to find an optimal task partitioning, allocation, and scheduling strategy.
- Uses **point-cloud compression** to reduce communication overhead for LiDAR data transmission.
- Real deployment reports about **34% lower localization latency** compared with OBU-only execution.

Mingyue Cui et al. “Offloading autonomous driving services via edge computing”. In: IEEE Internet of Things Journal 7. 10 (2020), pp. 10535–10547.

Autonomous Vehicles

C-V2X-enabled collaborative driving framework (CCAD)



CCAD framework overview. (a) C-V2X enabled a collaborative relationship between smart infrastructure and Autonomous vehicles. (b) C-V2X network stack for CCAD applications.(source)

Achieves about **15.5 ms C-V2X message latency** and approximately **98.8 ms end-to-end latency** in the collaborative lane-keeping case study.

➢ Key characteristics

- CCAD that connects vehicles and roadside infrastructure to **overcome the limited sensing range of single vehicles.**
- Relies on standardized C-V2X messages such as BSM, MAP, SPAT, TIM, and PVD instead of transmitting raw sensor data, reducing bandwidth requirements

Y He et al. "Towards C-V2X enabled collaborative autonomous driving". In: IEEE Transactions on Vehicular Technology 72. 12 (2023), pp. 15450–15462. DOI: 10.1109/ TVT.2023.3299844.

Autonomous Vehicles: Challenges

Computational overhead

Onboard compute can’t handle data-intensive services alone.

Privacy & security

Protect data in a dynamic vehicular environment.

Sensor data volume

Fuse huge streams from cameras, radar & LiDAR.

Energy & network

Resource allocation, energy use & network stability.

Summary

- IoT — lower latency with improved efficiency and security.
- Retail — personalized services and smarter inventory management.
- Autonomous vehicles — real-time decision-making at the edge.

Practice

1. What are the primary challenges of implementing edge computing in autonomous vehicles, and how are they addressed?

2. What are the key benefits of edge computing for IoT ecosystems, particularly in terms of latency and data security?

Project

Create a simulation model to compare the performance of edge computing vs. cloud computing in processing IoT data for autonomous vehicles.

➢ What you'll do

1. **Model the scenario** AVs generating sensor streams; define edge nodes (roadside units) vs. a cloud datacenter.
2. **Implement both paths** Route the same workloads through an edge pipeline and a cloud-only pipeline.
3. **Vary the conditions** Sweep data volume, network latency & task complexity.
4. **Compare results** Measure latency, throughput, bandwidth & reliability side by side.

➢ What to measure

- **Latency** End-to-end delay for a driving decision.
- **Bandwidth** Data volume pushed upstream in each design.
- **Throughput** Tasks handled per second under load.
- **Reliability** Behavior as the network degrades.

Example: in CloudSim/iFogSim, feed LiDAR frames through roadside-unit edge nodes vs. cloud-only; expect edge to win on latency as network delay grows.

References

- Gopika Premsankar, Mario Di Francesco, and Tarik Taleb. “Edge computing for the Internet of Things: A case study”. In: IEEE Internet of Things Journal 5. 2 (2018), pp. 1275–1284.
- Tie Qiu et al. “Edge computing in Industrial Internet of Things: Architecture, advances and challenges”. In: IEEE Communications Surveys & Tutorials 22. 4 (2020), pp. 2462–2488.
- J Lin et al. “An edge computing based public vehicle system for smart transportation”. In: IEEE Transactions on Vehicular Technology 69. 11 (2020), pp. 12635–12651. DOI: 10.1109/TVT.2020.3028497.
- Pawani Porambage et al. “Survey on multi-access edge computing for Internet of Things realization”. In: IEEE Communications Surveys & Tutorials 20. 4 (2018), pp. 2961–2991.
- Charith Perera et al. “Fog computing for sustainable smart cities: A survey”. In: ACM Computing Surveys (CSUR) 50. 3 (2017), pp. 1–43.
- M Saravanan and N C Srinidhi Srivatsan. “Intelligent communication between IoT devices on edges in retail sector”. In: Advances in Information and Communication Networks: Proceedings of the 2018 Future of Information and Communication Conference (FICC), Vol. 2. Springer, 2019, pp. 546–562.
- Shaoshan Liu et al. “Edge computing for autonomous driving: Opportunities and challenges”. In: Proceedings of the IEEE 107. 8 (2019), pp. 1697–1716.
- Qingyang Zhang et al. “OpenVDAP: An open vehicular data analytics platform for CAVs”. In: 2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS). IEEE, 2018, pp. 1310–1320.
- S Tang et al. “VECFrame: A vehicular edge computing framework for connected autonomous vehicles”. In: 2021 IEEE International Conference on Edge Computing (EDGE). 2021, pp. 68–77. DOI: 10.1109/EDGE53862.2021.00019.
- Sidi Lu, Yongtao Yao, and Weisong Shi. “Collaborative learning on the edges: A case study on connected vehicles”. In: 2nd USENIX Workshop on Hot Topics in Edge Computing (HotEdge 19), 2019.
- Mingyue Cui et al. “Offloading autonomous driving services via edge computing”. In: IEEE Internet of Things Journal 7. 10 (2020), pp. 10535–10547.
- B Wu et al. “To turn or not to turn, safecross is the answer”. In: 2022 IEEE 42nd International Conference on Distributed Computing Systems (ICDCS). 2022, pp. 414–424. DOI: 10.1109/ICDCS54860.2022.00047.
- Y He et al. “Towards C-V2X enabled collaborative autonomous driving”. In: IEEE Transactions on Vehicular Technology 72. 12 (2023), pp. 15450–15462. DOI: 10.1109/TVT.2023.3299844.