ORIGINAL ARTICLE

Real-Time Streaming Analytics and Latency Minimization in Autonomous Vehicle Big Data Pipelines

Dilshan Abeyratne[1]

[1] Uva Wellassa University, Department of Computer Science, Passara Road, Badulla, Sri Lanka.,

ABSTRACT

In modern autonomous vehicle ecosystems, massive volumes of sensor and contextual data are generated and analyzed almost continuously. Achieving real-time streaming analytics in this environment involves tackling stringent latency requirements and managing diverse data modalities, all while ensuring that the underlying infrastructure is both robust and scalable. Advanced data processing frameworks must accommodate high-frequency sensor readings, vehicular trajectory streams, and auxiliary contextual information, integrating them into a cohesive pipeline for intelligent decision-making. Latency minimization strategies increasingly rely on sophisticated data partitioning methods, parallel processing engines, and edge computing platforms, which bring computation closer to vehicles to alleviate bandwidth saturation and reduce delays. The interplay between data velocity, volume, and variety compels the adoption of cutting-edge solutions, including distributed message brokers, sliding-window analytics, and scalable machine learning models. These models incorporate matrix factorization, stochastic gradient updates, and complex transformations designed to extract meaningful features from continuous input streams. Ensuring timely response and reliable situational awareness requires careful attention to routing protocols, concurrency controls, and dynamic resource allocation. This paper explores the theoretical underpinnings of real-time analytics within autonomous vehicle pipelines and proposes strategies to minimize latency through optimized dataflows, adaptive scheduling algorithms, and secure communication channels. By addressing these critical facets, it underscores the necessity of a holistic, future-ready approach to real-time vehicular data processing.

Creative Commons License

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.
To view a copy of this license, visit https://creativecommons.org/licenses/by-nc/4.0/ or send a letter to Creative Commons, PO Box 1866, Mountain View, CA 94042, USA.

© Northern Reviews

1 | Introduction

Real-time streaming analytics in autonomous vehicle ecosystems demands rigorous approaches to data handling, fusion, and dissemination [1]. Vehicles, operating as both data sources and real-time decision-making platforms, generate constant streams from high-definition cameras, LiDARs, radars, GPS sensors, and onboard control units [2]. These data streams must be processed promptly to enable functions such as collision avoidance, path planning, and dynamic traffic management. Given the proliferation of connected vehicles and smart infrastructures, the volume of data grows at exponential rates, highlighting the need for specialized systems capable of scaling efficiently while adhering to near-zero latency requirements [3]. Among the challenges is the management of diverse data types: structured telematics data, semi-structured sensor logs, and unstructured images or point clouds. Handling the fusion of these heterogeneous streams in real time demands sophisticated frameworks that can ingest, normalize, and analyze data at sub-second latencies. [4]

Contemporary approaches to vehicular big data pipelines leverage distributed computing paradigms where streaming frameworks, such as Apache Kafka or distributed in-memory processing systems, orchestrate data inflows [5]. These systems use partitioned data streams, parallel scheduling strategies, and in-network computing concepts to ensure that insights are produced within milliseconds of data arrival. As such, the interplay between advanced networking protocols (e.g., 5G or dedicated short-range communications) and powerful edge computing platforms becomes paramount in guaranteeing timeliness of analytics [6]. Nevertheless, while distributed systems can reduce computational overhead on individual vehicles, they also introduce additional complexities regarding synchronization, concurrency, and fault tolerance. Latency minimization is central to ensuring safety in autonomous contexts [7]. Early detection of obstacles, rapid hazard mitigation, and the ability to adapt to evolving road conditions necessitate swift data-to-decision workflows. Typical constraints may involve end-to-end latencies of under 100 milliseconds, imposing strict computational, networking, and memory requirements [8]. Real-time analytics pipelines commonly integrate specialized hardware accelerators, including GPUs or TPUs, which execute machine learning and computer vision tasks in parallel [9]. Furthermore, advanced concurrency control protocols coordinate parallel tasks so that partial results are combined seamlessly without data races or consistency issues.

Mathematical models underpin these decisions by framing problem spaces in terms of matrix operations, vector transformations, and multidimensional probability distributions [10]. For instance, the continuous flow of sensor data can be represented as a time-varying matrix $\mathbf{X}(t)$, where each column reflects a sensor modality and each row reflects a temporal snapshot. Various transformations, from wavelet decompositions to low-rank approximations, can then be employed to extract latent features. The interplay of linear algebra with real-time constraints becomes more pronounced when combining multiple streams of data into an integrated representation that informs predictive models for obstacle detection and trajectory estimation. [11]

The reliability of autonomous vehicle operations depends on factors beyond raw computational throughput [12]. Issues of data integrity, fault tolerance, load balancing, and secure communication are equally significant. Fault tolerance often hinges on replication strategies, leader-election mechanisms, and robust checkpointing routines [13]. On the security side, real-time systems must implement cryptographic protocols and secure enclaves without incurring additional latency. As the number of connected vehicles and edge devices grows, these considerations become inextricable from discussions of resource allocation, admission control, and prioritization [14]. Consequently, advanced research in network architectures, distributed machine learning, and real-time systems converges on constructing integrated solutions, balancing throughput demands with the imperative to minimize end-to-end latency for safety-critical decisions.

2 | Theoretical Foundations of Real-Time Analytics in Autonomous Vehicle Ecosystems

Real-time analytics in autonomous vehicle ecosystems can be framed as a set of specialized operators acting on continuous dataflows [15]. From a functional viewpoint, one can define the data stream as a mapping $\mathbf{x} : \mathbb{T} \to \mathbb{R}^n$, where $\mathbb{T}$ represents discrete or continuous time indices, and $\mathbb{R}^n$ denotes an n-dimensional space capturing multimodal sensor data. The transformation of interest might involve feature extraction, prediction, or control feedback [16]. The concept of stream processing entails evaluating a chain of transformations $\mathcal{F}_1, \mathcal{F}_2, \ldots, \mathcal{F}_k$, applied sequentially or in parallel to generate timely insights. The cardinal objective is minimizing the time lag between data ingestion and actionable output.

Essential to the success of these transformations is the notion of operator micro-batching, sliding windows, or tumbling windows, which partition the data stream into manageable segments [17]. A typical example arises when a streaming system implements a window function $W(\mathbf{x}(t), \Delta t)$ that accumulates data over an interval Δt to perform localized computations. This approach allows for resource optimization and parallelization, as computations can be distributed across multiple processing nodes. The underlying scheduling algorithm may follow a round-robin assignment or a work-stealing paradigm, each having distinct implications for load balancing and system throughput [18]. Hence, the theoretical foundations of real-time analytics blend scheduling theory with the design of parallelizable dataflow operators, ensuring that the delays from queuing, processing, and communication are kept within acceptable bounds. [19]

Linear algebra plays a pivotal role in advanced autonomous vehicle analytics. Consider a scenario where onboard and roadside sensors collectively produce high-dimensional data vectors $\mathbf{z}_i$ at each time step i. Real-time anomaly detection can be carried out through statistical methods that track the deviation of incoming vectors from a dynamic mean $\boldsymbol{\mu}(t)$ or covariance matrix $\Sigma(t)$. One might define an evolving Mahalanobis distance $\delta^2 = (\mathbf{z}_i - \boldsymbol{\mu}(t))^T \Sigma(t)^{-1} (\mathbf{z}_i - \boldsymbol{\mu}(t))$, signaling potential anomalies if δ^2 exceeds a threshold derived from theoretical distributions. Such matrix computations, repeated over numerous parallel streams, push the limits of real-time system design. [20]

Further theoretical insights emerge from queuing theory, which delineates how tasks accumulate and get serviced within the computational pipeline. Let λ be the arrival rate of data items and μ the service rate in a single-server model [21]. In real-world autonomous vehicle networks, however, multiple servers, each with distinct service rates and scheduling policies, operate concurrently. This distributed environment necessitates expansions of classical queuing models to accommodate multi-tier topologies and dynamic resource scaling [22]. One might adopt open or closed network models, employing product-form solutions to approximate system congestion [23]. The result is a framework that links arrival rates, service rates, and buffer capacities to the probability of incurring excessive delay.

In certain real-time scenarios, robust estimation methods are employed to tackle uncertainties in streaming data [24]. Stochastic filtering techniques like the Kalman filter or particle filter can be extended to large-scale sensor networks. An augmented state vector might contain not only position and velocity estimates but also system parameters characterizing sensor reliability [25]. Updating these estimates in near-real-time relies on matrix multiplication, inversion, and addition steps that must be optimized to handle the sheer velocity of data [26]. GPU-accelerated libraries, specialized hardware instructions, or quantum-inspired algorithms can further reduce the computational latency.

At the intersection of control theory and real-time analytics lies the concept of dynamic feedback loops [27]. Vehicles continuously sense their environment, update internal states using filtering algorithms, and make control decisions. The system can be modeled as a continuous-time dynamical system expressed by $\dot{\mathbf{x}}(t) = A\mathbf{x}(t) + B\mathbf{u}(t)$, where $\mathbf{x}(t)$ encapsulates the vehicle's states (e.g., velocity, orientation) and $\mathbf{u}(t)$ is the control input derived from the streaming analytics pipeline. Stability, controllability, and observability considerations underscore the importance of timely data processing: any delay in the real-time feedback mechanism can degrade system performance or lead to catastrophic failures [28]. Solving the corresponding Riccati equations or linear quadratic regulator (LQR) problems on the fly exemplifies how advanced mathematics underlies the entire pipeline.

3 | Latency Minimization Strategies and Performance Metrics

Minimizing latency in the context of autonomous vehicle big data pipelines is a critical and multifaceted challenge [29]. These pipelines process vast quantities of heterogeneous data in real-time, sourced from LiDAR sensors, radar modules, GPS, onboard cameras, and vehicular telemetry [30]. This real-time nature imposes stringent requirements on system responsiveness, and even slight latencies can compromise safety, reliability, or functional accuracy. Thus, latency minimization is not just a performance optimization but a safety-critical requirement. [31]

3.1 Architectural Approaches to Latency Reduction

The architectural foundation of a data pipeline has profound implications for its latency profile. Traditional centralized cloud-based models are often inadequate due to the delay introduced by wide-area communication [32]. Instead, modern pipelines are trending toward decentralized and hierarchical architectures, with edge computing forming the first line of compute offloading. [33]

In such a topology, compute resources are distributed closer to the data sources—either onboard the vehicle or at nearby edge servers. This architectural shift is motivated by the need to minimize round-trip times (RTT) and propagation delays [34]. When processing is performed at the edge, raw data need not traverse the network to distant cloud infrastructures, thereby significantly reducing latency.

This offloading can be formalized using an optimization model [35]. Let N denote the set of computational nodes, including edge, fog, and cloud resources. Each node $i \in N$ is characterized by a computational capacity C_i, and the latency associated with routing data between nodes i and j is denoted by ℓ_{ij}. A central objective is to assign each task from the sensor stream to a compute node such that the total latency is minimized, while ensuring that no node is overloaded beyond its capacity. [36]

Mathematically, the optimization problem may be represented as: [37]

$$\min_{\{x_{ij}\}} \sum_{i \in S} \sum_{j \in N} x_{ij} \cdot \ell_{ij}$$

subject to:

$$\sum_{i \in S} x_{ij} \cdot r_i \leq C_j, \quad \forall j \in N$$

$$x_{ij} \in \{0, 1\}, \quad \forall i \in S, j \in N$$

Here, S is the set of sensor-generated tasks, r_i is the resource requirement of task i, and x_{ij} is a binary variable indicating if task i is assigned to node j.

3.2 Latency as a Composite Metric

Latency in autonomous systems is rarely a singular metric [38]. It is instead an aggregate of several time components: queuing delays at processing nodes (τ_q), computational or processing times (τ_p), and communication latencies (τ_c). The end-to-end latency τ for a task can be approximated as: [39]

$$\tau = \tau_q + \tau_p + \tau_c[40]$$

The goal of latency minimization strategies is to ensure that τ consistently remains below a predetermined threshold τ_{th}, which is critical for the correct and timely operation of autonomous driving functions such as obstacle detection, trajectory planning, and real-time decision-making.

Furthermore, latency metrics are often supplemented with percentile-based measures. For example, the 99th percentile latency (τ_{99}) is used to characterize tail latency—ensuring that even the slowest 1% of data points are processed within acceptable time frames. This is crucial in safety-critical environments where occasional delays can have catastrophic consequences. [41]

3.3 Load Balancing and Dynamic Resource Allocation

Adaptive load balancing is an essential tool in maintaining low latency under varying system loads. It involves distributing tasks or data streams across multiple processing units such that no single resource becomes a bottleneck [42]. Load balancing can be static—based on historical data—or dynamic—responsive to real-time measurements of system state.

In mathematical terms, if $\mathbf{d} \in \mathbb{R}^n$ is the task distribution vector and $\mathbf{t} \in \mathbb{R}^n$ the corresponding processing time vector at each node, one might seek to minimize:

$$\min \|\mathbf{d} \cdot \mathbf{t}\|_1$$

subject to constraints on each node's processing capacity: [43]

$$d_i \cdot t_i \leq C_i, \quad \forall i[44]$$

Dynamic load balancing systems often implement feedback loops, where node metrics such as CPU usage, memory availability, and current queue lengths are periodically reported to a centralized controller or decentralized agents. These measurements are then used to redistribute workload in near real-time. [45]

A prominent heuristic used in such scenarios is the least-loaded-first (LLF) approach, where tasks are preferentially assigned to the least-burdened node. More sophisticated strategies use predictive models trained on historical performance metrics to anticipate load surges and proactively allocate resources. [46]

3.4 Stream Partitioning Techniques

Partitioning streaming data into smaller chunks that can be processed in parallel is a well-established strategy to improve throughput and reduce latency [47]. Each partition is typically mapped to a separate processing instance, which allows for horizontal scaling. However, poor partitioning can lead to data skew—where certain partitions contain significantly more data than others, leading to processing delays

[48]. To address this, partitioning must be data-aware. Common techniques include: [49]

- **Hash Partitioning:** Uses a hash function to assign data records to partitions. Ensures uniform distribution for data with well-behaved keys.
- **Range Partitioning:** Divides data based on key ranges. Effective for ordered data but sensitive to key distributions. [50]
- **Dynamic Partitioning:** Periodically reevaluates and reshuffles partition boundaries based on runtime statistics.

These strategies often require a careful trade-off between parallelism and inter-node communication overhead, particularly when re-partitioning data mid-stream. [51]

3.5 Fault Tolerance with Latency Constraints

Latency minimization must coexist with fault tolerance—a critical requirement in autonomous systems. Traditional approaches, such as full replication of data streams and processing state, can introduce significant overhead and thus negatively impact latency. [52]
Erasure coding provides a latency-efficient alternative. Instead of full duplication, data is split into fragments and encoded with redundancy [53]. During a failure, only a subset of these fragments is required to reconstruct the original data [54]. This minimizes storage and transmission overhead but introduces decoding delay.
Fast recovery mechanisms rely on incremental checkpointing, where only changes to the processing state are logged [55]. Upon failure, the system rolls back to the last known good state and resumes processing with minimal recomputation. Checkpoint frequency is a tunable parameter: higher frequency reduces recovery time but increases I/O load. [56]
Let δ_r be the time to reassign tasks post-failure, and δ_c the time to restore from the checkpoint. The failover latency τ_f must satisfy: [57]

$$\tau_f = \delta_r + \delta_c \leq \tau_{th}$$

where τ_{th} is the maximum tolerable latency for uninterrupted operations.

3.6 Monitoring and Performance Tuning

Continuous monitoring is essential to latency-aware system design [58]. Streaming systems must be instrumented to collect metrics such as task arrival rates, operator execution times, and queue lengths. These metrics feed into control mechanisms that adjust system parameters in real-time. [59]
Predictive analytics play an increasingly important role in latency management. Time-series forecasting models, often based on ARIMA, LSTM, or Transformer architectures, are used to predict load patterns [60]. Based on predictions, systems can scale resources, increase parallelism, or pre-warm operators to handle anticipated surges. [61]
Tuning the performance of such pipelines involves solving a multi-objective optimization problem, where latency minimization must be balanced against energy efficiency, resource utilization, and throughput. Let $\mathcal{L}$ denote latency, $\mathcal{E}$ energy, and $\mathcal{T}$ throughput. A composite objective might be: [62]

$$\min\left(w_1 \cdot \mathcal{L} + w_2 \cdot \mathcal{E} - w_3 \cdot \mathcal{T}\right)$$

where w_1, w_2, w_3 are application-specific weights.

3.7 End-to-End Pipeline Coordination

One of the greatest challenges in latency minimization lies in the coordination of end-to-end pipeline stages—from data ingestion to final actuation [63]. Autonomous vehicles typically involve a pipeline comprising perception, localization, mapping, path planning, and control. Latency in one stage propagates downstream, potentially compounding delays. [64]
Thus, coordination mechanisms must include: [65]

- **Pipelined Execution:** Ensuring overlapping execution of stages to minimize idle time.
- **Backpressure Management:** Regulating data flow to prevent upstream operators from overwhelming downstream stages.
- **Cross-layer Optimization:** Jointly optimizing network routing, compute scheduling, and task allocation.

To model dependencies between stages, directed acyclic graphs (DAGs) are often employed, with each node representing a computation and edges denoting data flow. Minimizing latency becomes a problem of minimizing the critical path in the DAG under capacity constraints. [66]

Minimizing latency in autonomous vehicle big data pipelines is a grand challenge that intersects systems design, networking, optimization, and machine learning. From architectural decisions like edge computing to algorithmic solutions such as adaptive partitioning and dynamic load balancing, a holistic approach is necessary [67]. Moreover, the pursuit of lower latency must always be harmonized with other design constraints such as fault tolerance, energy efficiency, and safety guarantees [68]. Only through integrated, adaptive, and predictive strategies can such systems achieve the sub-second responsiveness essential for real-time autonomy.

4 | Distributed and Parallel Processing Architectures for Autonomous Vehicle Data

The architectural design of big data pipelines in autonomous vehicle systems commonly adopts a layered approach that includes edge devices (vehicle sensors), edge servers (roadside units or local microdata centers), and cloud or central data centers [69]. Each layer handles different parts of the computation pipeline to minimize data movement, optimize latency, and ensure resilience. From a mathematical perspective, the integration of distributed systems can be viewed through graph-theoretical constructs [70]. Consider a graph $G = (V, E)$ where vertices V denote processing elements (edge or cloud servers) and edges E represent communication links with associated bandwidth and latency attributes. Assigning tasks to vertices in a manner that respects capacity while minimizing data transfer times becomes a graph partitioning or min-cut problem. [71]

Parallel processing frameworks within these architectures typically employ the map-reduce or dataflow paradigm [72]. At a high level, streaming data is split into shards or micro-batches, mapped to parallel executors that perform computations, and then reduced or aggregated to produce global insights. In a streaming context, the reduce phase may be replaced by continuous operators that maintain rolling states [73]. Since autonomous vehicle data can be high-dimensional, containing streaming point clouds, images, or telemetry, these operators are often GPU-accelerated to handle large matrix manipulations efficiently. Decomposition strategies are used to leverage parallel threads or kernels, for example splitting a matrix $\mathbf{M}$ into sub-blocks that can be processed in tandem.

Cluster configuration for such systems involves multiple nodes, each equipped with CPUs, GPUs, or specialized accelerators like FPGAs [74]. Scheduling policies must account for the heterogeneous nature of hardware, matching workloads to the resources where they execute most effectively [75]. Queuing models are generalized to incorporate heterogeneous service rates, $\mu^{(1)}, \mu^{(2)}, \ldots, \mu^{(k)}$, for different resource types. The overarching objective is to guarantee that the effective arrival rate λ does not saturate any subset of these resources, thus preventing local bottlenecks that increase latency. Simultaneously, oversubscription scenarios can arise in which tasks queue at GPU-equipped nodes while CPU nodes remain underutilized [76]. Advanced solutions might involve dynamic reconfiguration of tasks to exploit idle resources, provided the overhead of transferring partial states does not outweigh the gains.

Another critical aspect is the continuum from in-vehicle processing to the cloud [77]. Vehicles themselves may have onboard computing capabilities powerful enough to handle first-stage processing, especially for urgent tasks like obstacle detection or sensor fusion needed for immediate control decisions. This local analysis can be encapsulated in a partial result vector $\mathbf{p}$, which is then transmitted to edge servers for further aggregation. Reducing the dimensionality of data before transmission, for instance by compressing sensor readings or extracting key features using convolutional neural networks, lessens network demands [78]. The trade-off emerges in deciding how much computation to delegate to vehicles versus offloading to more robust edge or cloud infrastructure [79]. Formalizing this requires solving an optimization problem that balances local processing energy costs, available bandwidth, and latency constraints.

Fault tolerance must be integrated into parallel processing architectures [80]. Autonomous vehicles cannot afford system failures that interrupt essential services. Such resilience may be achieved through replication strategies where each task runs concurrently on two or more nodes, or through checkpoint-based methods that periodically save the state of streaming operators [81]. Graphically, if tasks are mapped to a subgraph $H \subseteq G$, replication would imply mapping them to two node-disjoint subgraphs H_1 and H_2 with minimal overlap, so that hardware failures in one subgraph do not affect the other [82]. While replication increases reliability, it also increases resource usage and can contribute to higher overall latencies. Solutions must calibrate the level of replication to the criticality of each application,

reflecting a spectrum of risk tolerance in real-time analytics tasks. [83]
In many instances, the pipeline includes a machine learning component that must be continuously updated with streaming data. Online learning algorithms, which update model parameters $\mathbf{w}$ incrementally, are well-suited here. A typical online gradient descent approach may adjust parameters as $\mathbf{w}_{t+1} = \mathbf{w}_t - \alpha \nabla L(\mathbf{w}_t; \mathbf{x}_t)$, where L is a loss function and $\mathbf{x}_t$ is the data batch at time t. If these updates occur in a distributed fashion, partial gradients from different nodes must be aggregated, requiring an all-reduce communication step [84]. Balancing the frequency of communication-intensive synchronization with the desire for up-to-date models is an ongoing research challenge. In latency-sensitive contexts, asynchronous updates may be employed, allowing local nodes to proceed with partially stale parameters while waiting for global updates [85]. This yields faster iteration times but can cause convergence issues if data distributions shift quickly or if staleness becomes too pronounced [86]. Consequently, system designers often rely on advanced theoretical results from distributed optimization that bound the trade-off between synchronization intervals and convergence guarantees under streaming conditions.

5 | Security and Privacy Considerations in Real-Time Autonomous Systems

Security and privacy concerns are paramount in real-time autonomous vehicle ecosystems, especially when large amounts of data are being collected and analyzed [87]. Unauthorized access to sensor streams or tampering with data pipelines can lead to incorrect decisions, with severe consequences for both safety and privacy. Cryptographic protocols are commonly employed to secure communications between vehicles, edge nodes, and the cloud [88]. Symmetric encryption mechanisms such as AES can be integrated into the dataflow, although the key distribution process must be carefully orchestrated to prevent bottlenecks [89]. For end-to-end security, public key infrastructure (PKI) ensures that only authorized entities can send valid data, while ephemeral key agreements, like Diffie-Hellman, can frequently refresh session keys to minimize exposure.
Latency, however, is a limiting factor in security protocol design [90]. Computational overhead from encryption and decryption processes must not undermine the real-time constraints. One can frame this as a minimal overhead optimization problem: if γ represents the cryptographic latency per data chunk and τ is the unencrypted data processing time, the total overhead ratio γ/τ should remain below a set threshold [91]. Hardware-accelerated cryptographic modules can help reduce γ by offloading encryption tasks to specialized co-processors. Similarly, lightweight encryption schemes tailored for resource-constrained systems, such as elliptical curve cryptography, can provide security without excessive computational costs. [92]
Privacy entails strict controls over how data is accessed and utilized [93]. Many sensor data streams contain sensitive information about drivers' locations, habits, or even visual data of surroundings. When aggregated, these data points form comprehensive profiles that risk exposing personally identifiable information [94]. Anonymization techniques can be incorporated into the pipeline, such as k-anonymity or differential privacy, although each introduces potential latency overhead. Differential privacy mechanisms, for instance, inject noise into aggregated statistics to shield individual contributions [95]. In streaming contexts, noise-injection is dynamic, but doing so in real time requires additional computations [96]. If ν represents the magnitude of noise added and ϵ the privacy budget, the system must ensure minimal distortion of crucial features, thus balancing the trade-off between data utility and confidentiality.
A complementary line of defense is secure multi-party computation (MPC), where computations on encrypted data are performed without directly revealing the underlying inputs [97]. A simplified conceptualization of MPC might define an encrypted data vector $\tilde{\mathbf{x}}_i$ at node i, with partial decryptions requiring multiple key shares from different nodes. While this approach bolsters security, it introduces overhead in orchestrating partial decryptions, especially in a latency-sensitive environment. To mitigate these constraints, advanced cryptographic primitives, like fully homomorphic encryption, are sometimes considered, though practical usage remains limited by significant performance costs. [98]
Access control systems represent another layer of security. They determine who is permitted to read sensor data, issue commands to actuators, or modify system configurations [99]. Role-based access control (RBAC) or attribute-based access control (ABAC) systems are commonly integrated into the pipeline [100]. Each request is evaluated against a policy that weighs the requester's identity, context, or security clearances. The microsecond-level overhead from these checks must be factored into the real-time analytics

design [101]. Although these overheads are generally small, the cumulative effect over millions of requests can be nontrivial. Overly stringent policy checks can also introduce new points of congestion [102]. Consequently, designing streamlined policy engines with caching or partial evaluation is vital for preserving low-latency operations. [103]

Regulatory frameworks further complicate security and privacy requirements. Compliance with standards like ISO 21434 or data protection regulations influences how data is stored, transmitted, and processed, including mandatory logging and auditing mechanisms [104]. Audit logs allow detection and analysis of security breaches but require constant writing to secure storage. If we let κ be the average time to log an event, the cumulative delays grow with the logging frequency [105]. Solutions range from selective logging to compressed logging, where only essential data is recorded in detail. The interplay between regulatory demands and real-time constraints can necessitate architectural trade-offs, including adopting specialized hardware modules dedicated to logging or cryptographic operations. [106]

6 | Conclusion

Autonomous vehicle ecosystems demand real-time streaming analytics that can reliably process immense volumes of heterogeneous data with stringent latency requirements [107]. The interplay between advanced processing frameworks, network topologies, and computational hardware reveals a complex landscape where mathematical models—spanning linear algebra, queuing theory, and distributed optimization—provide the underpinnings for robust pipeline design. Equally vital are the architectural strategies that place computation strategically across edge, fog, and cloud layers, reducing end-to-end delays [108]. Parallel processing, dynamic task scheduling, and load balancing emerge as powerful instruments in maintaining sub-second response times despite high data velocity and unpredictable workloads.

The necessity of secure and private data handling adds further dimensions to the challenge [109]. Encryption protocols must be chosen judiciously to avoid becoming bottlenecks [110]. Distributed key management, privacy-preserving techniques, and role-based access controls must interlock seamlessly with real-time analytics. Emerging cryptographic methods, along with hardware acceleration, push the boundaries of what is feasible in latency-sensitive applications [111]. This comprehensive interplay of latency minimization, parallel architectures, and security underscores the multidisciplinary nature of modern autonomous vehicle pipelines.

Future solutions may incorporate breakthroughs in quantum computing or specialized AI accelerators, opening avenues for faster matrix manipulations and enhanced security protocols [112]. Advances in distributed machine learning could reduce synchronization overhead and provide even more rapid adaptation to changing traffic or sensor conditions. By harnessing the strengths of these innovations and building on a firm theoretical foundation, the industry can continue to refine the safety, efficiency, and responsiveness of autonomous vehicle systems in an era of ever-expanding data streams. [113]

References

[1] M. Farisco, K. Evers, and A. Salles, "Towards establishing criteria for the ethical analysis of artificial intelligence.," *Science and engineering ethics*, vol. 26, pp. 2413–2425, 7 2020.

[2] W. Chanlongrat, T. Apichanapong, P. Sinngam, and W. Chaisangmongkon, "A semi-automated system for person re-identification adaptation to cross-outfit and cross-posture scenarios," *Applied Intelligence*, vol. 52, pp. 9501–9520, 1 2022.

[3] K. Wang, L. Zhang, Q. Xia, L. Pu, and J. Chen, "Cross-domain learning using optimized pseudo labels: toward adaptive car detection in different weather conditions and urban cities," *Neural Computing and Applications*, vol. 34, pp. 4519–4529, 10 2021.

[4] R. Guidotti, A. Monreale, S. Ruggieri, F. Naretto, F. Turini, D. Pedreschi, and F. Giannotti, "Stable and actionable explanations of black-box models through factual and counterfactual rules," *Data Mining and Knowledge Discovery*, vol. 38, pp. 2825–2862, 11 2022.

[5] D. N. Wagner, "Economic patterns in a world with artificial intelligence," *Evolutionary and Institutional Economics Review*, vol. 17, pp. 111–131, 1 2020.

[6] L. Yang, H. Jiang, Q. Song, and J. Guo, "A survey on long-tailed visual recognition," *International Journal of Computer Vision*, vol. 130, pp. 1837–1872, 5 2022.

[7] R. Alt, "Electronic markets on robotics," *Electronic Markets*, vol. 31, pp. 465–471, 9 2021.

[8] V. Goding and K. Tranter, "The robot and human futures: Visualising autonomy in law and science fiction," *Law and Critique*, vol. 34, pp. 315–340, 9 2023.

[9] R. Kaur, G. Karmakar, F. Xia, and M. Imran, "Deep learning: survey of environmental and camera impacts on internet of things images.," *Artificial intelligence review*, vol. 56, pp. 1–9638, 2 2023.

[10] E. H. Houssein, A. Hammad, and A. A. Ali, "Human emotion recognition from eeg-based brain–computer interface using machine learning: a comprehensive review," *Neural Computing and Applications*, vol. 34, pp. 12527–12557, 5 2022.

[11] C. Zhang, W. Huang, T. Niu, Z. Liu, G. Li, and D. Cao, "Review of clustering technology and its application in coordinating vehicle subsystems," *Automotive Innovation*, 1 2023.

[12] Y. Kawahara, "Understanding the underlying algorithms and theories of machine learning," *Impact*, vol. 2020, pp. 40–42, 4 2020.

[13] S. Doncieux, R. Chatila, S. Straube, and F. Kirchner, "Human-centered ai and robotics," *AI Perspectives*, vol. 4, 1 2022.

[14] A. Staino, A. Suwalka, P. Mitra, and B. Basu, "Real-time detection and recognition of railway traffic signals using deep learning," *Journal of Big Data Analytics in Transportation*, vol. 4, pp. 57–71, 6 2022.

[15] Y. Liu, Y. Luo, and A. M. Naidech, "Big data in stroke: How to use big data to make the next management decision.," *Neurotherapeutics : the journal of the American Society for Experimental NeuroTherapeutics*, vol. 20, pp. 744–757, 3 2023.

[16] I. Boškoski, B. Orlandini, L. G. Papparella, M. V. Matteo, M. D. Siena, V. Pontecorvi, and G. Costamagna, "Robotics and artificial intelligence in gastrointestinal endoscopy: Updated review of the literature and state of the art," *Current Robotics Reports*, vol. 2, pp. 43–54, 2 2021.

[17] Z. Ahmad, S. Rahim, M. Zubair, and J. Abdul-Ghafar, "Artificial intelligence (ai) in medicine, current applications and future role with special emphasis on its potential and promise in pathology: present and future impact, obstacles including costs and acceptance among pathologists, practical and philosophical considerations. a comprehensive review.," *Diagnostic pathology*, vol. 16, pp. 24–24, 3 2021.

[18] M. G. V. Kumar, S. Karunakaran, S. Chandre, R. K. Godi, P. Manirajkumar, and A. Balaram, "Implementation of microgrid digital twin system for unmanned vehicles with cloud computing techniques," *SN Computer Science*, vol. 4, 7 2023.

[19] T. Zhang, J. Zou, and W. Jia, "Fast and robust road sign detection in driver assistance systems," *Applied Intelligence*, vol. 48, pp. 4113–4127, 5 2018.

[20] L. Cultrera, F. Becattini, L. Seidenari, P. Pala, and A. D. Bimbo, "Explaining autonomous driving with visual attention and end-to-end trainable region proposals," *Journal of Ambient Intelligence and Humanized Computing*, 2 2023.

[21] D. Belter, J. Wietrzykowski, and P. Skrzypczyński, "Employing natural terrain semantics in motion planning for a multi-legged robot," *Journal of Intelligent & Robotic Systems*, vol. 93, pp. 723–743, 5 2018.

[22] P. Henz, "Ethical and legal responsibility for artificial intelligence," *Discover Artificial Intelligence*, vol. 1, pp. 1–5, 9 2021.

[23] R. Bautista-Montesano, R. Bustamante-Bello, and R. A. Ramirez-Mendoza, "Explainable navigation system using fuzzy reinforcement learning," *International Journal on Interactive Design and Manufacturing (IJIDeM)*, vol. 14, pp. 1411–1428, 10 2020.

[24] A. Radovic, M. Williams, D. Rousseau, M. Kagan, D. Bonacorsi, A. Himmel, A. Aurisano, K. Terao, and T. Wongjirad, "Machine learning at the energy and intensity frontiers of particle physics.," *Nature*, vol. 560, pp. 41–48, 8 2018.

[25] D. Proserpio, J. Hauser, X. Liu, T. Amano, A. Burnap, T. Guo, D. Lee, R. A. Lewis, K. Misra, E. Schwarz, A. Timoshenko, L. Xu, and H. Yoganarasimhan, "Soul and machine (learning)," *Marketing Letters*, vol. 31, pp. 393–404, 8 2020.

[26] M. Abouelyazid, "Adversarial deep reinforcement learning to mitigate sensor and communication attacks for secure swarm robotics," *Journal of Intelligent Connectivity and Emerging Technologies*, vol. 8, no. 3, pp. 94–112, 2023.

[27] A. L. Thandu, V. S. Thommandru, and P. Gera, "Data science in healthcare monitoring under covid-19 detection by extended hybrid leader-based compressed neural network," *New Generation Computing*, vol. 41, pp. 669–696, 6 2023.

[28] M. K. Sana, Z. Hussain, A. Shah, and M. H. Maqsood, "Artificial intelligence in celiac disease.," *Computers in biology and medicine*, vol. 125, pp. 103996–, 9 2020.

[29] D. Giveki and M. Karami, "Scene classification using a new radial basis function classifier and integrated sift–lbp features," *Pattern Analysis and Applications*, vol. 23, pp. 1071–1084, 2 2020.

[30] J. Zhang and Z.-M. Zhang, "Ethics and governance of trustworthy medical artificial intelligence.," *BMC medical informatics and decision making*, vol. 23, pp. 7–, 1 2023.

[31] E. Arkin, N. Yadikar, X. Xu, A. Aysa, and K. Ubul, "A survey: object detection methods from cnn to transformer," *Multimedia Tools and Applications*, vol. 82, pp. 21353–21383, 10 2022.

[32] S. E. H. Kiashari, A. Nahvi, H. Bakhoda, A. Homayounfard, and M. Tashakori, "Evaluation of driver drowsiness using respiration analysis by thermal imaging on a driving simulator," *Multimedia Tools and Applications*, vol. 79, pp. 17793–17815, 2 2020.

[33] L. Alzubaidi, J. Zhang, A. J. Humaidi, A. Q. Al-Dujaili, Y. Duan, O. Al-Shamma, J. Santamaría, M. A. Fadhel, M. Al-Amidie, and L. Farhan, "Review of deep learning: concepts, cnn architectures, challenges, applications, future directions," *Journal of big data*, vol. 8, pp. 1–74, 3 2021.

[34] I. G. Daza, R. Izquierdo, L. M. Martínez, O. Benderius, and D. F. Llorca, "Sim-to-real transfer and reality gap modeling in model predictive control for autonomous driving," *Applied Intelligence*, vol. 53, pp. 12719–12735, 10 2022.

[35] M. Abouelyazid, "Comparative evaluation of sort, deepsort, and bytetrack for multiple object tracking in highway videos," *International Journal of Sustainable Infrastructure for Cities and Societies*, vol. 8, no. 11, pp. 42–52, 2023.

[36] H. Naderi, L. Goli, and S. Kasaei, "Generating unrestricted adversarial examples via three parameteres," *Multimedia Tools and Applications*, vol. 81, pp. 21919–21938, 3 2022.

[37] D. W. McShea, "Evolutionary trends and goal directedness.," *Synthese*, vol. 201, pp. 178–, 5 2023.

[38] J. F. P. Crespo and A. Wichert, "Reinforcement learning applied to games," *SN Applied Sciences*, vol. 2, pp. 1–16, 4 2020.

[39] C.-A. Brust, C. Käding, and J. Denzler, "Active and incremental learning with weak supervision," *KI - Künstliche Intelligenz*, vol. 34, pp. 165–180, 1 2020.

[40] S. H. Alsamhi, O. Ma, and M. S. Ansari, "Convergence of machine learning and robotics communication in collaborative assembly: Mobility, connectivity and future perspectives," *Journal of Intelligent & Robotic Systems*, vol. 98, pp. 541–566, 10 2019.

[41] M. A. Zulkifley, S. R. Abdani, and N. H. Zulkifley, "Pterygium-net: a deep learning approach to pterygium detection and localization," *Multimedia Tools and Applications*, vol. 78, pp. 34563–34584, 8 2019.

[42] R. Drainville and F. Vis, "Elephant motorbikes and too many neckties: epistemic spatialization as a framework for investigating patterns of bias in convolutional neural networks," *AI & SOCIETY*, vol. 39, pp. 1079–1093, 8 2022.

[43] H. Sun, W. Ni, P.-L. Teh, and A. Sohal, "An agent-mediated and entrepreneurship-oriented model for assessing high technology transfer," *Information Technology and Management*, vol. 26, pp. 19–38, 5 2023.

[44] M. Shafique, T. Theocharides, H. H. Li, and C. J. Xue, "Introduction to the special issue on accelerating ai on the edge – part 2," *ACM Transactions on Embedded Computing Systems*, vol. 21, pp. 1–5, 11 2022.

[45] S. Dhakal, Q. Chen, D. Qu, D. Carillo, Q. Yang, and S. Fu, "Sniffer faster r-cnn: A joint camera-lidar object detection framework with proposal refinement," in *2023 IEEE International Conference on Mobility, Operations, Services and Technologies (MOST)*, pp. 1–10, IEEE, 2023.

[46] Z. Ouyang, J. Niu, Y. Liu, and M. Guizani, "Deep cnn-based real-time traffic light detector

for self-driving vehicles," *IEEE Transactions on Mobile Computing*, vol. 19, pp. 300–313, 2 2020.

[47] N. G. Rezk, A.-F. Attia, M. A. El-Rashidy, A. El-Sayed, and E. E.-D. Hemdan, "An efficient plant disease recognition system using hybrid convolutional neural networks (cnns) and conditional random fields (crfs) for smart iot applications in agriculture," *International Journal of Computational Intelligence Systems*, vol. 15, 8 2022.

[48] P. Kim, H. Lee, and H. J. Kim, "Autonomous flight with robust visual odometry under dynamic lighting conditions," *Autonomous Robots*, vol. 43, pp. 1605–1622, 10 2018.

[49] H. Lyre, "The state space of artificial intelligence," *Minds and Machines*, vol. 30, pp. 325–347, 9 2020.

[50] Y. Tkachenko and K. Jedidi, "A megastudy on the predictability of personal information from facial images: Disentangling demographic and non-demographic signals.," *Scientific reports*, vol. 13, pp. 21073–, 11 2023.

[51] G. Kyprianou, L. Doitsidis, and S. A. Chatzichristofis, "Towards the achievement of path planning with multi-robot systems in dynamic environments," *Journal of Intelligent & Robotic Systems*, vol. 104, 12 2021.

[52] M. Muniswamaiah, T. Agerwala, and C. C. Tappert, "Federated query processing for big data in data science," in *2019 IEEE International Conference on Big Data (Big Data)*, pp. 6145–6147, IEEE, 2019.

[53] Q. Lin, T. Li, P. M. Shakeel, and R. D. J. Samuel, "Advanced artificial intelligence in heart rate and blood pressure monitoring for stress management," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, pp. 3329–3340, 11 2020.

[54] Y. Liao, K. Dijkstra, and R. A. Zwaan, "Linguistic and non-linguistic cues in motion event endpoint description: The selection between english to and towards.," *Memory & cognition*, vol. 51, pp. 982–996, 11 2022.

[55] O. Goodall, "The rural enterprise crime complex: 'undefendable rural space' and the threat from the fortress farm," *Crime, Law and Social Change*, vol. 80, pp. 215–235, 2 2023.

[56] A. Saha and B. C. Dhara, "3d lidar-based obstacle detection and tracking for autonomous navigation in dynamic environments," *International Journal of Intelligent Robotics and Applications*, vol. 8, pp. 39–60, 11 2023.

[57] M. Humenberger, Y. Cabon, N. Pion, P. Weinzaepfel, D. Lee, N. Guérin, T. Sattler, and G. Csurka, "Investigating the role of image retrieval for visual localization," *International Journal of Computer Vision*, vol. 130, pp. 1811–1836, 5 2022.

[58] C. Polat, "Great unclear expectations: A study on the defectiveness of autonomous systems," *Journal of Consumer Policy*, vol. 46, pp. 363–386, 6 2023.

[59] A. Rasheed, A. I. Umar, S. H. Shirazi, Z. Khan, and M. Shahzad, "Cover-based multiple book genre recognition using an improved multimodal network," *International Journal on Document Analysis and Recognition (IJDAR)*, vol. 26, pp. 65–88, 9 2022.

[60] M. Muniswamaiah, T. Agerwala, and C. C. Tappert, "Federated query processing for big data in data science," in *2019 IEEE International Conference on Big Data (Big Data)*, pp. 6145–6147, IEEE, 2019.

[61] S. Dhakal, D. Qu, D. Carrillo, Q. Yang, and S. Fu, "Oasd: An open approach to self-driving vehicle," in *2021 Fourth International Conference on Connected and Autonomous Driving (MetroCAD)*, pp. 54–61, IEEE, 2021.

[62] M. S. Rais, K. Zouaidia, and R. Boudour, "Enhanced decision making in multi-scenarios for autonomous vehicles using alternative bidirectional q network," *Neural Computing and Applications*, vol. 34, pp. 15981–15996, 5 2022.

[63] P. Dumouchel, "Ethics & robotics, embodiment and vulnerability," *International Journal of Social Robotics*, vol. 15, pp. 2087–2099, 3 2022.

[64] E. O. Arkhangelskaya and S. I. Nikolenko, "Deep learning for natural language processing: A survey," *Journal of Mathematical Sciences*, vol. 273, pp. 533–582, 6 2023.

[65] D. C. Le and A. N. Zincir-Heywood, "A frontier: Dependable, reliable and secure machine learning for network/system management," *Journal of Network and Systems Management*, vol. 28, pp. 827–849, 1 2020.

[66] Z. Li and A. Zhou, "Self-selection salient region-based scene recognition using slight-weight convolutional neural network," *Journal of Intelligent & Robotic Systems*, vol. 102, pp. 1–16, 6 2021.

[67] M. Nadin, "Machine intelligence: a chimera," *AI & SOCIETY*, vol. 34, pp. 215–242, 4 2018.

[68] S. P. Patro, N. Padhy, and D. Chiranjevi, "Ambient assisted living predictive model for cardiovascular disease prediction using supervised learning," *Evolutionary Intelligence*, vol. 14, pp. 941–969, 9 2020.

[69] M. Leodolter, C. Plant, and N. Brändle, "Rotation invariant gps trajectory mining," *GeoInformatica*, vol. 28, pp. 89–115, 6 2023.

[70] M. Muniswamaiah, T. Agerwala, and C. Tappert, "Data virtualization for analytics and business intelligence in big data," in *CS & IT Conference Proceedings*, vol. 9, CS & IT Conference Proceedings, 2019.

[71] A. van Teijlingen, T. Tuttle, H. Bouchachia, B. Sathian, and E. van Teijlingen, "Artificial intelligence and health in nepal.," *Nepal journal of epidemiology*, vol. 10, pp. 915–918, 9 2020.

[72] M. Kaveh and M. S. Mesgari, "Application of meta-heuristic algorithms for training neural networks and deep learning architectures: A comprehensive review.," *Neural processing letters*, vol. 55, pp. 1–4622, 10 2022.

[73] M. R. Babavalian and K. Kiani, "Learning distribution of video captions using conditional gan," *Multimedia Tools and Applications*, vol. 83, pp. 9137–9159, 6 2023.

[74] P. A. Shah, M. K. Srinath, R. Gayathri, P. Puvandran, and S. K. Selvaraj, "Advanced solid-state welding based on computational manufacturing using the additive manufacturing process," *International Journal on Interactive Design and Manufacturing (IJIDeM)*, 3 2023.

[75] L. Agarwal and B. Verma, "From methods to datasets: A survey on image-caption generators," *Multimedia Tools and Applications*, vol. 83, pp. 28077–28123, 8 2023.

[76] E. Marchisio, "In support of "no-fault" civil liability rules for artificial intelligence," *SN social sciences*, vol. 1, pp. 54–, 1 2021.

[77] S. A. Matveev, I. V. Oseledets, E. Ponomarev, and A. Chertkov, "Overview of visualization methods for artificial neural networks," *Computational Mathematics and Mathematical Physics*, vol. 61, pp. 887–899, 7 2021.

[78] A. Filotheou, A. Tzitzis, E. Tsardoulias, A. Dimitriou, A. Symeonidis, G. Sergiadis, and L. Petrou, "Passive global localisation of mobile robot via 2d fourier-mellin invariant matching," *Journal of Intelligent & Robotic Systems*, vol. 104, 1 2022.

[79] R. Nayak, U. C. Pati, and S. K. Das, "A comprehensive review of datasets for detection and localization of video anomalies: a step towards data-centric artificial intelligence-based video anomaly detection," *Multimedia Tools and Applications*, vol. 83, pp. 59617–59674, 12 2023.

[80] N. Nagaraj, H. L. Gururaj, B. H. Swathi, and Y.-C. Hu, "Passenger flow prediction in bus transportation system using deep learning.," *Multimedia tools and applications*, vol. 81, pp. 12519–12542, 2 2022.

[81] M. Lee, K. Y. Han, J. Yu, and Y.-S. Lee, "A new lane following method based on deep learning for automated vehicles using surround view images," *Journal of Ambient Intelligence and Humanized Computing*, vol. 14, pp. 1–14, 9 2019.

[82] F. Niederman and E. W. Baker, "Ethics and ai issues: Old container with new wine?," *Information Systems Frontiers*, vol. 25, pp. 9–28, 6 2022.

[83] Y. Song, R. Fan, S. Huang, Z. Zhu, and R. Tong, "A three-stage real-time detector for traffic signs in large panoramas," *Computational Visual Media*, vol. 5, pp. 403–416, 9 2019.

[84] J. Xu, Z. Lu, and Y. Xie, "Loan default prediction of chinese p2p market: a machine learning methodology.," *Scientific reports*, vol. 11, pp. 18759–18759, 9 2021.

[85] B. A. El-Rahiem, M. Amin, A. Sedik, F. E. A. E. Samie, and A. M. Iliyasu, "An efficient multi-biometric cancellable biometric scheme based on deep fusion and deep dream," *Journal of ambient intelligence and humanized computing*, vol. 13, pp. 1–13, 11 2021.

[86] X. Li, Y. Zhang, and D. Kong, "$\mathrm{E}^2 - pv - rcnn : improving 3d object detection via enhancing keypoint features,'' A$ $-35874, 62022$.

[87] S. Laxmi, S. K. Gupta, and S. Kumar, "Intuitionistic fuzzy proximal support vector machine for multicategory classification problems," *Soft Computing*, vol. 25, pp. 14039–14057, 9 2021.

[88] M. Mirbabaie, L. Hofeditz, N. R. J. Frick, and S. Stieglitz, "Artificial intelligence in hospitals: providing a status quo of ethical considerations in academia to guide future research.," *AI & society*, vol. 37, pp. 1–22, 6 2021.

[89] J. Li, H. Pan, L. Wu, C. Huang, X. Luo, and Y. Xu, "Class-guided human motion prediction via multi-spatial-temporal supervision," *Neural Computing and Applications*, vol. 35, pp. 9463–9479, 3 2023.

[90] L. Zhang, F. Jiang, B. Kong, J. Yang, and C. Wang, "Real-time lane detection by using biologically inspired attention mechanism to learn contextual information," *Cognitive Computation*, vol. 13, pp. 1333–1344, 9 2021.

[91] C. Gandrez, F. Mantelet, A. Aoussat, and F. Jeremie, "Predict the risk feeling for drivers of autonomous cars: an application of deep learning methods," *International Journal on Interactive Design and Manufacturing (IJIDeM)*, vol. 17, pp. 249–259, 9 2022.

[92] N. Zaghari, M. Fathy, S. M. Jameii, M. Sabokrou, and M. Shahverdy, "Improving the learning of self-driving vehicles based on real driving behavior using deep neural network techniques," *The Journal of Supercomputing*, vol. 77, pp. 3752–3794, 8 2020.

[93] K. Seetharam, S. Raina, and P. P. Sengupta, "The role of artificial intelligence in echocardiography.," *Current cardiology reports*, vol. 22, pp. 99–99, 7 2020.

[94] N. Sharma and N. Jindal, "Emerging artificial intelligence applications: metaverse, iot, cybersecurity, healthcare - an overview," *Multimedia Tools and Applications*, vol. 83, pp. 57317–57345, 12 2023.

[95] O. Lappi, J. Pekkanen, P. Rinkkala, S. Tuhkanen, A. J. Tuononen, and J.-P. Virtanen, "Humans use optokinetic eye movements to track waypoints for steering," *Scientific reports*, vol. 10, pp. 4175–, 3 2020.

[96] L. Yang, Y. Liu, H. Yu, X. Fang, L. Song, D. Li, and C. Yingyi, "Computer vision models in intelligent aquaculture with emphasis on fish detection and behavior analysis: A review," *Archives of Computational Methods in Engineering*, vol. 28, pp. 2785–2816, 9 2020.

[97] M. Muniswamaiah, T. Agerwala, and C. Tappert, "Big data in cloud computing review and opportunities," *arXiv preprint arXiv:1912.10821*, 2019.

[98] A. Saxena and P. Mishra, "Optimizing electric vehicle energy management systems with a hybrid lstm-cnn architecture," *Tensorgate Journal of Sustainable*, 2022.

[99] W. Christensen and K. Bicknell, "Cognitive control, intentions, and problem solving in skill learning.," *Synthese*, vol. 200, pp. 460–, 11 2022.

[100] B. H. Leso, M. N. Cortimiglia, A. Ghezzi, and V. Minatogawa, "Exploring digital transformation capability via a blended perspective of dynamic capabilities and digital maturity: a pattern matching approach," *Review of Managerial Science*, vol. 18, pp. 1149–1187, 9 2023.

[101] C. Shorten, T. M. Khoshgoftaar, and B. Furht, "Deep learning applications for covid-19," *Journal of big data*, vol. 8, pp. 1–54, 1 2021.

[102] D. Humphreys, A. Kupresanin, M. D. Boyer, J. Canik, C.-S. Chang, E. C. Cyr, R. Granetz, J. Hittinger, E. Kolemen, E. Lawrence, V. Pascucci, A. Patra, and D. Schissel, "Advancing fusion with machine learning research needs workshop report," *Journal of Fusion Energy*, vol. 39, pp. 123–155, 9 2020.

[103] null Irfanullah, T. Hussain, A. Iqbal, B. Yang, and A. Hussain, "Retracted article: Real time violence detection in surveillance videos using convolutional neural networks," *Multimedia Tools and Applications*, vol. 81, pp. 38151–38173, 4 2022.

[104] S. Khatri, H. Vachhani, S. Shah, J. Bhatia, M. Chaturvedi, S. Tanwar, and N. Kumar, "Machine learning models and techniques for vanet based traffic management: Implementation issues and challenges," *Peer-to-Peer Networking and Applications*, vol. 14, pp. 1778–1805, 9 2020.

[105] S.-C. Chen, "Multimedia for autonomous driving," *IEEE MultiMedia*, vol. 26, pp. 5–8, 7 2019.

[106] H. Sheikh, "European digital sovereignty: A layered approach," *Digital Society*, vol. 1, 11 2022.

[107] R. Yang and S. Wibowo, "User trust in artificial intelligence: A comprehensive conceptual framework," *Electronic Markets*, vol. 32, pp. 2053–2077, 11 2022.

[108] M. Aqib, D. Kumar, and S. Tripathi, "Machine learning for fog computing: Review, opportunities and a fog application classifier and scheduler," *Wireless Personal Communications*, vol. 129, pp. 853–880, 12 2022.

[109] J. Kim, Y. Choi, M. Park, S. Lee, and S. Kim, "Multi-sensor-based detection and tracking of moving

objects for relative position estimation in autonomous driving conditions," *The Journal of Supercomputing*, vol. 76, pp. 8225–8247, 3 2019.

[110] J. Cowls, A. Tsamados, M. Taddeo, and L. Floridi, "The ai gambit: leveraging artificial intelligence to combat climate change-opportunities, challenges, and recommendations.," *AI & society*, vol. 38, pp. 1–25, 10 2021.

[111] M. Mulvenna, R. Bond, J. Delaney, F. M. Dawoodbhoy, J. Boger, C. Potts, and R. Turkington, "Ethical issues in democratizing digital phenotypes and machine learning in the next generation of digital health technologies," *Philosophy & technology*, vol. 34, pp. 1–16, 3 2021.

[112] J. F. Oeding, R. J. Williams, B. U. Nwachukwu, R. K. Martin, B. T. Kelly, J. Karlsson, C. L. Camp, A. D. Pearle, A. S. Ranawat, and A. Pareek, "A practical guide to the development and deployment of deep learning models for the orthopedic surgeon: part i.," *Knee surgery, sports traumatology, arthroscopy : official journal of the ESSKA*, vol. 31, pp. 382–389, 11 2022.

[113] Éloi Zablocki, H. Ben-Younes, P. Pérez, and M. Cord, "Explainability of deep vision-based autonomous driving systems: Review and challenges," *International Journal of Computer Vision*, vol. 130, pp. 2425–2452, 8 2022.