Hardware-Aware Optimization of Vision-Language Models for Resource-Constrained Autonomous Vehicles

Albert Cearponter
Department of Computer Science, University of Houston, Houston, TX, USA.
albertwork@uh.edu

Uday J. Reddy
Department of Computer Science, University of Central Florida, Orlando, FL, USA.
ujreddy@ucf.edu

Aarav A. Ghosh
Department of Computer Science, Binghamton University, Binghamton, NY, USA.
aarav790@binghamton.edu

Abstract

Vision-language models have emerged as powerful instruments for unified perception, scene understanding, and reasoning in autonomous systems. Their ability to align visual and textual representations enables zero-shot semantic interpretation, natural-language interaction, and richer uncertainty communication. However, deployment in resource-constrained autonomous vehicles introduces stringent constraints on latency, memory, energy, thermal budget, hardware heterogeneity, and safety certification. This paper presents a system-level analysis of hardware-aware optimization for vision-language models in automotive contexts, treating the problem not merely as model compression but as a cross-layer challenge involving architecture design, runtime infrastructure, safety assurance, sustainability, and governance. The analysis examines structural trade-offs among pruning, quantization, knowledge distillation, and efficient backbone selection, and discusses how these interact with accelerator capabilities, memory hierarchies, and real-time scheduling. The discussion emphasizes that optimization decisions cannot be reduced to accuracy-throughput curves; they must account for degradation under distribution shift, redundancy requirements, fairness implications, energy accountability, and compliance with emerging regulatory frameworks. Drawing on developments in efficient deep learning, multi-modal perception, and edge computing, the paper provides an interdisciplinary perspective on how hardware-aware vision-language model deployment can support dependable, equitable, and sustainable autonomous mobility. The paper also highlights the importance of transparent reporting, institutional oversight, and cross-domain collaboration between automotive engineering, machine learning, and policy communities.

Keywords

vision-language models; hardware-aware optimization; autonomous vehicles; edge computing; model compression; safety assurance; sustainability; AI governance.

1. Introduction

The integration of vision-language models into autonomous vehicle architectures represents a significant shift from conventional modular perception pipelines toward systems that jointly reason over visual scenes and natural language. The foundational attention mechanism [1] and subsequent vision transformer formulations [2] demonstrated that self-attention can model long-range dependencies in visual data, while contrastive language-image pretraining [3] showed that aligned embedding spaces allow flexible downstream interpretation without extensive task-specific retraining. In automotive contexts, these capabilities can support scene captioning, zero-shot object recognition, and contextual anomaly description, thereby offering new forms of semantic awareness for planning and human-machine interaction. Nevertheless, autonomous vehicles differ fundamentally from cloud-based artificial intelligence systems. They must operate under severe energy, latency, memory, thermal, and reliability constraints while moving through dynamic environments. This requires hardware-aware optimization to be considered as a systemic property of the entire deployment stack.

Early end-to-end learning systems for self-driving cars illustrated the potential of directly mapping visual inputs to control commands, while also exposing vulnerabilities related to dataset bias, interpretability, and validation [4]. Subsequent research has produced more comprehensive surveys of deep learning methods for autonomous driving [5], alongside multimodal benchmark suites such as KITTI [6] and nuScenes [7] that support standardized evaluation of detection, tracking, and prediction across diverse conditions. These resources have shaped the evaluation culture of the field, but they also reveal a persistent gap between laboratory performance and the reliability required for real-time edge deployment. This gap is magnified for vision-language models, which often inherit large parameter counts and memory footprints from cloud training. Hardware-aware optimization, therefore, must address architectural selection, numerical precision, scheduling, and safety governance simultaneously. The remainder of this paper develops that argument through a system-level lens.

2. Vision-Language Models in Autonomous Driving: System Requirements and Constraints

Vision-language models used in autonomous driving typically combine visual encoders, textual encoders, and cross-modal fusion modules. Residual convolutional networks remain important for low-level feature extraction and real-time perception because of their stable optimization and efficient hardware mappings [8]. Although transformer-based visual backbones have become more common, the choice of backbone exerts strong influence over hardware compatibility, bandwidth utilization, and cache behavior. Automotive platforms are composed of heterogeneous system-on-chip devices, digital signal processors, graphics processing units, neural processing units, and safety islands. These components have different memory hierarchies, quantization support, and deterministic execution guarantees. A model that performs well on a data-center accelerator may exhibit severe utilization imbalances when ported to an automotive system-on-chip, where dynamic voltage and frequency scaling, shared memory contention, and thermal throttling introduce non-negligible variability.

The operational requirements of autonomous vehicles are frequently expressed through end-to-end latency budgets, worst-case execution time envelopes, and energy caps per kilometer traveled. Vision-language inference often lies on the critical path for tasks such as question answering about roadside signage, summarization of unusual events, and natural-language communication with remote operators. These tasks are less periodic than object detection or trajectory prediction, yet they can consume substantial compute when triggered. Resource-

constrained deployment therefore demands careful partitioning between always-on perception models, event-triggered language decoding, and low-power standby modes. In addition, safety standards require that sensor-to-actuator chains remain deterministic and inspectable. This clashes with autoregressive decoding patterns and attention over long sequences, whose runtime can vary with input complexity and token length. Hardware-aware optimization must thus impose architectural constraints that are not typical in general-purpose vision-language research.

From a system engineering perspective, the memory footprint is often a more binding constraint than raw arithmetic throughput. Large weight matrices create pressure on off-chip memory bandwidth, while activation maps and attention keys and values can exceed on-chip scratchpad capacity. This leads to frequent data movement across the memory hierarchy, increasing energy consumption and making latency less predictable. Hardware-aware methods must therefore be evaluated not only by multiply-accumulate reductions but also by their effect on memory traffic, data reuse, and parallelism. The combination of vision transformers, language decoders, and cross-attention layers creates complex dataflow patterns that can be especially difficult to schedule on edge accelerators. These structural considerations motivate the cross-layer optimization discussed in the next section.

3. Hardware-Aware Optimization as a Cross-Layer Design Problem

Hardware-aware optimization is often framed narrowly as model compression, but a more accurate systems perspective treats it as a joint search over algorithmic structure, numerical representation, data layout, and accelerator mapping. Classical work on deep compression demonstrated that pruning, quantization, and coding can substantially reduce model size without proportional accuracy loss [9]. Integer-arithmetic-only inference further showed that careful quantization-aware training can preserve performance while enabling efficient deployment on fixed-point hardware [10]. Knowledge distillation provides an additional mechanism for transferring capabilities from large teacher models to smaller student models, compressing not only parameters but also representational invariances [11]. In automotive contexts, these methods need to be applied under explicit latency and memory budgets because deployment decisions directly influence safety margins and energy consumption.

Efficient architectural families such as MobileNetV2 [12] and EfficientNet [13] illustrate how design choices such as inverted residuals, linear bottlenecks, and compound scaling can produce favorable trade-offs between accuracy and computational cost. However, vision-language models introduce cross-modal fusion layers that do not always map cleanly onto such efficient patterns. Cross-attention between visual tokens and text tokens can generate dynamic memory access, variable sequence lengths, and irregular parallelism. Hardware-aware optimization therefore requires co-design of the model architecture and the compiler scheduler, rather than applying post hoc compression to a fixed model. Techniques such as structured pruning, mixed-precision quantization, and operator fusion must account for the memory bandwidth and systolic array dimensions of the target accelerator. A technical report on an edge-native text-to-image foundation model trained entirely on domestic accelerators highlights the feasibility of training and deploying cross-modal foundation models outside conventional data-center hardware assumptions [14]. This development reinforces the view that hardware-aware model design is not exclusively a hyperscaler-led trajectory, but rather an evolving global ecosystem with distinct architectural constraints and incentives.

A cross-layer design approach also implies that algorithmic choices and runtime policies cannot be separated. For example, a quantized student model may have low average latency

but exhibit worst-case token decoding schedules that violate automotive deadlines when rare linguistic constructs appear. Similarly, activation sparsity may reduce arithmetic but increase divergence across parallel processing elements, causing synchronization overhead. System designers must therefore consider not only static metrics such as parameter count and floating-point operations, but also dynamic metrics such as tail latency, memory contention, and thermal dissipation under sustained use. These trade-offs cannot be captured by a single Pareto frontier because the objective functions differ across stakeholders. Vehicle manufacturers may prioritize certification cost, fleet operators may prioritize energy efficiency, and regulators may prioritize explainability and resilience. Hardware-aware optimization is thus a governance problem as much as a technical optimization problem.

4. Architectural Trade-Offs and Edge-Native Deployment

The architectural design space for vision-language models in autonomous vehicles can be organized around three interacting axes: backbone efficiency, cross-modal fusion location, and decoding strategy. In early fusion designs, visual and textual representations are combined before deep processing, which can reduce intermediate activation sizes but may sacrifice modality-specific structure. In late fusion designs, separate encoders produce high-level embeddings that are aligned through a lightweight interaction module, which is often more compatible with pre-existing perception stacks. Hybrid designs place cross-attention at selected scales, enabling linguistic context to influence visual interpretation without processing every low-level feature through language-conditioned layers. Each choice affects memory traffic, cache residency, and the feasibility of operator fusion on edge accelerators. Hardware-aware deployment often favors asymmetric designs in which the visual encoder runs on a deterministic neural processing unit and the language decoder runs on a general-purpose processor with flexible control flow.

Pruning, quantization, and distillation are not interchangeable. Unstructured pruning can reduce model size significantly but may produce irregular sparsity that is difficult to exploit on structured accelerators. Structured pruning removes entire channels or attention heads, preserving dense matrix operations and simplifying compilation, but it may remove functional diversity needed for rare scene understanding. Quantization reduces bit width and memory bandwidth, yet its robustness under extreme lighting, adversarial inputs, and sensor degradation must be validated explicitly. Knowledge distillation can help a compact student model approximate the teacher's soft decisions, but the teacher's biases can also be inherited. Hardware-aware optimization should therefore combine these methods in a staged process that includes sensitivity analysis, robustness evaluation, and continuous monitoring. This process is not a one-time engineering exercise but an ongoing lifecycle practice because automotive fleets operate across changing geographic, climatic, and social conditions.

Edge-native deployment further introduces heterogeneity across vehicle generations and regional hardware ecosystems. Some platforms emphasize high-throughput matrix units, while others optimize for sparse tensor acceleration or low-power digital signal processing. A model optimized for one platform may underutilize another, leading to uneven fleet performance and complicating over-the-air updates. The emergence of edge-native foundation models trained on China-developed accelerators [14] illustrates that platform-specific training can reduce dependency on imported hardware, but it also raises questions about interoperability, standardization, and verification across different automotive markets. From a socio-technical perspective, hardware-aware optimization is therefore entangled with supply chain resilience, technology sovereignty, and regional regulatory divergence. A robust

deployment strategy must account for these institutional dimensions rather than assuming a single global hardware target.

5. Runtime Infrastructure and Scheduling for Resource-Constrained Platforms

Runtime infrastructure determines how vision-language inference interacts with conventional perception, planning, and control workloads. Automotive systems increasingly use middleware that manages priority levels, memory partitions, and hardware accelerators under real-time constraints. A vision-language model task may be activated by an unusual-object detector, by remote operator request, or by a passenger query. Such event-triggered workloads are difficult to schedule because their execution time depends on visual token count, language complexity, and decoding length. Middleware must therefore provide admission control, preemption, and graceful degradation. For example, the system may first run a lightweight captioning model to triage an event; only when the preliminary confidence is low may it invoke a larger vision-language reasoner. This layered approach reduces average energy but creates complex coupling between false positives, latency, and task priority inversion.

The energy implications of such scheduling decisions are not neutral. Large-scale neural network training already carries significant carbon and energy costs [15], and the broader policy literature has shown that deep learning workloads in language processing can impose substantial environmental burdens when deployed at scale [16]. Although automotive edge inference is smaller in individual footprint, fleet-level effects become material because millions of vehicles may operate for years. Energy-aware scheduling can dynamically adjust precision, context length, and model capacity according to route complexity, remaining battery, and availability of offloading infrastructure. Such policies must be transparent and auditable, especially when they affect safety-critical semantic understanding in low-power modes.

Infrastructure also includes data logging, model update distribution, and remote supervision. Vision-language models can generate natural-language summaries of unusual scenes that are valuable for fleet incident review and regulatory audits. However, transmitting and storing such summaries raises privacy, consent, and data minimization concerns. Edge-native architectures may keep sensitive visual data on-vehicle and transmit only abstractions, but the linguistic summaries themselves can reveal location, identity, or vulnerable situations. Runtime infrastructure must therefore enforce policy controls such as differential privacy, access logging, and geographic data localization. Hardware-aware optimization affects these controls because smaller models that run locally can reduce the incentive to offload data, thereby supporting privacy by architecture. The interaction between energy efficiency, latency, and data governance demonstrates that resource-constrained vision-language model deployment cannot be understood as a purely computational problem.

6. Robustness, Redundancy, and Safety Assurance

Deep learning systems have enabled hierarchical feature learning across perception, language, and control tasks [17], but their deployment in safety-critical settings requires explicit treatment of distribution shift, specification ambiguity, and adversarial perturbation. The broader artificial intelligence safety literature identifies problems such as negative side effects, reward hacking, and safe exploration that are highly relevant to autonomous driving even when no reinforcement learning is involved [18]. For vision-language models, robustness failures may be expressed through misleading captions, incorrect visual grounding, or harmful dialogue with passengers. Such failures may not be captured by aggregate accuracy metrics

because they are rare, context-dependent, and semantically consequential. Safety assurance therefore demands layered redundancy, where vision-language model outputs are cross-checked against deterministic perception modules, map priors, and physical constraints.

Multi-modal perception in autonomous driving has been studied extensively, with deep architectures fusing camera, lidar, and radar for detection and semantic segmentation [19]. These sensor fusion approaches offer useful lessons for vision-language model robustness because they show how complementary modalities can compensate for individual sensor failures. Similarly, class imbalance and rare-object detection have motivated loss functions such as focal loss to focus learning on hard examples [20]. In a vision-language system, analogous issues arise when rare linguistic queries refer to uncommon objects or unusual traffic situations. Hardware-aware compression may disproportionately degrade performance on rare classes if these classes depend on subtler feature interactions. Robustness evaluation must therefore include adversarial scenes, rare vocabulary, dialectal variation, and degraded visual conditions such as rain, fog, glare, and partial occlusion.

Safety assurance for resource-constrained vision-language models should be organized around risk-based tiers. High-risk functions, such as interpreting traffic control signals or emergency instructions, require higher redundancy, stricter latency bounds, and more conservative model capacity. Lower-risk functions, such as conversational route commentary, can tolerate larger variability and may exploit cloud offloading. This tiering should be enforced at runtime by a safety monitor that evaluates confidence, semantic consistency, and agreement with deterministic perception. The monitor itself must be small, verifiable, and isolated from the learned model. Hardware-aware optimization can support safety by reducing model complexity to a level where formal verification and exhaustive testing become more feasible. However, reducing complexity should not substitute for fail-safe mechanisms. A compressed vision-language model may still produce fluent but incorrect outputs, so the system must treat language generation as an advisory channel rather than an authoritative control signal unless additional validation is present.

7. Sustainability and Energy Governance

Sustainability considerations extend beyond individual vehicles to the entire life cycle of model development, fleet deployment, and hardwareHardware-Aware Optimization of Vision-Language Models for Resource-Constrained Autonomous Vehicles

Abstract

Vision-language models have emerged as powerful instruments for unified perception, scene understanding, and reasoning in autonomous systems. Their ability to align visual and textual representations enables zero-shot semantic interpretation, natural-language interaction, and richer uncertainty communication. However, deployment in resource-constrained autonomous vehicles introduces stringent constraints on latency, memory, energy, thermal budget, hardware heterogeneity, and safety certification. This paper presents a system-level analysis of hardware-aware optimization for vision-language models in automotive contexts, treating the problem not merely as model compression but as a cross-layer challenge involving architecture design, runtime infrastructure, safety assurance, sustainability, and governance. The analysis examines structural trade-offs among pruning, quantization, knowledge distillation, and efficient backbone selection, and discusses how these interact with accelerator capabilities, memory hierarchies, and real-time scheduling. The discussion emphasizes that optimization decisions cannot be reduced to accuracy-throughput curves;

they must account for degradation under distribution shift, redundancy requirements, fairness implications, energy accountability, and compliance with emerging regulatory frameworks. Drawing on developments in efficient deep learning, multi-modal perception, and edge computing, the paper provides an interdisciplinary perspective on how hardware-aware vision-language model deployment can support dependable, equitable, and sustainable autonomous mobility. The paper also highlights the importance of transparent reporting, institutional oversight, and cross-domain collaboration between automotive engineering, machine learning, and policy communities.

Keywords

vision-language models; hardware-aware optimization; autonomous vehicles; edge computing; model compression; safety assurance; sustainability; AI governance

1. Introduction

The integration of vision-language models into autonomous vehicle architectures represents a significant shift from conventional modular perception pipelines toward systems that jointly reason over visual scenes and natural language. The foundational attention mechanism [1] and subsequent vision transformer formulations [2] demonstrated that self-attention can model long-range dependencies in visual data, while contrastive language-image pretraining [3] showed that aligned embedding spaces allow flexible downstream interpretation without extensive task-specific retraining. In automotive contexts, these capabilities can support scene captioning, zero-shot object recognition, and contextual anomaly description, thereby offering new forms of semantic awareness for planning and human-machine interaction. Nevertheless, autonomous vehicles differ fundamentally from cloud-based artificial intelligence systems. They must operate under severe energy, latency, memory, thermal, and reliability constraints while moving through dynamic environments. This requires hardware-aware optimization to be considered as a systemic property of the entire deployment stack.

Early end-to-end learning systems for self-driving cars illustrated the potential of directly mapping visual inputs to control commands, while also exposing vulnerabilities related to dataset bias, interpretability, and validation [4]. Subsequent research has produced more comprehensive surveys of deep learning methods for autonomous driving [5], alongside multimodal benchmark suites such as KITTI [6] and nuScenes [7] that support standardized evaluation of detection, tracking, and prediction across diverse conditions. These resources have shaped the evaluation culture of the field, but they also reveal a persistent gap between laboratory performance and the reliability required for real-time edge deployment. This gap is magnified for vision-language models, which often inherit large parameter counts and memory footprints from cloud training. Hardware-aware optimization, therefore, must address architectural selection, numerical precision, scheduling, and safety governance simultaneously. The remainder of this paper develops that argument through a system-level lens.

2. Vision-Language Models in Autonomous Driving: System Requirements and Constraints

Vision-language models used in autonomous driving typically combine visual encoders, textual encoders, and cross-modal fusion modules. Residual convolutional networks remain important for low-level feature extraction and real-time perception because of their stable optimization and efficient hardware mappings [8]. Although transformer-based visual backbones have become more common, the choice of backbone exerts strong influence over

hardware compatibility, bandwidth utilization, and cache behavior. Automotive platforms are composed of heterogeneous system-on-chip devices, digital signal processors, graphics processing units, neural processing units, and safety islands. These components have different memory hierarchies, quantization support, and deterministic execution guarantees. A model that performs well on a data-center accelerator may exhibit severe utilization imbalances when ported to an automotive system-on-chip, where dynamic voltage and frequency scaling, shared memory contention, and thermal throttling introduce non-negligible variability.

The operational requirements of autonomous vehicles are frequently expressed through end-to-end latency budgets, worst-case execution time envelopes, and energy caps per kilometer traveled. Vision-language inference often lies on the critical path for tasks such as question answering about roadside signage, summarization of unusual events, and natural-language communication with remote operators. These tasks are less periodic than object detection or trajectory prediction, yet they can consume substantial compute when triggered. Resource-constrained deployment therefore demands careful partitioning between always-on perception models, event-triggered language decoding, and low-power standby modes. In addition, safety standards require that sensor-to-actuator chains remain deterministic and inspectable. This clashes with autoregressive decoding patterns and attention over long sequences, whose runtime can vary with input complexity and token length. Hardware-aware optimization must thus impose architectural constraints that are not typical in general-purpose vision-language research.

From a system engineering perspective, the memory footprint is often a more binding constraint than raw arithmetic throughput. Large weight matrices create pressure on off-chip memory bandwidth, while activation maps and attention keys and values can exceed on-chip scratchpad capacity. This leads to frequent data movement across the memory hierarchy, increasing energy consumption and making latency less predictable. Hardware-aware methods must therefore be evaluated not only by multiply-accumulate reductions but also by their effect on memory traffic, data reuse, and parallelism. The combination of vision transformers, language decoders, and cross-attention layers creates complex dataflow patterns that can be especially difficult to schedule on edge accelerators. These structural considerations motivate the cross-layer optimization discussed in the next section.

3. Hardware-Aware Optimization as a Cross-Layer Design Problem

Hardware-aware optimization is often framed narrowly as model compression, but a more accurate systems perspective treats it as a joint search over algorithmic structure, numerical representation, data layout, and accelerator mapping. Classical work on deep compression demonstrated that pruning, quantization, and coding can substantially reduce model size without proportional accuracy loss [9]. Integer-arithmetic-only inference further showed that careful quantization-aware training can preserve performance while enabling efficient deployment on fixed-point hardware [10]. Knowledge distillation provides an additional mechanism for transferring capabilities from large teacher models to smaller student models, compressing not only parameters but also representational invariances [11]. In automotive contexts, these methods need to be applied under explicit latency and memory budgets because deployment decisions directly influence safety margins and energy consumption.

Efficient architectural families such as MobileNetV2 [12] and EfficientNet [13] illustrate how design choices such as inverted residuals, linear bottlenecks, and compound scaling can produce favorable trade-offs between accuracy and computational cost. However, vision-language models introduce cross-modal fusion layers that do not always map cleanly onto

such efficient patterns. Cross-attention between visual tokens and text tokens can generate dynamic memory access, variable sequence lengths, and irregular parallelism. Hardware-aware optimization therefore requires co-design of the model architecture and the compiler scheduler, rather than applying post hoc compression to a fixed model. Techniques such as structured pruning, mixed-precision quantization, and operator fusion must account for the memory bandwidth and systolic array dimensions of the target accelerator. A technical report on an edge-native text-to-image foundation model trained entirely on domestic accelerators highlights the feasibility of training and deploying cross-modal foundation models outside conventional data-center hardware assumptions [14]. This development reinforces the view that hardware-aware model design is not exclusively a hyperscaler-led trajectory, but rather an evolving global ecosystem with distinct architectural constraints and incentives.

A cross-layer design approach also implies that algorithmic choices and runtime policies cannot be separated. For example, a quantized student model may have low average latency but exhibit worst-case token decoding schedules that violate automotive deadlines when rare linguistic constructs appear. Similarly, activation sparsity may reduce arithmetic but increase divergence across parallel processing elements, causing synchronization overhead. System designers must therefore consider not only static metrics such as parameter count and floating-point operations, but also dynamic metrics such as tail latency, memory contention, and thermal dissipation under sustained use. These trade-offs cannot be captured by a single Pareto frontier because the objective functions differ across stakeholders. Vehicle manufacturers may prioritize certification cost, fleet operators may prioritize energy efficiency, and regulators may prioritize explainability and resilience. Hardware-aware optimization is thus a governance problem as much as a technical optimization problem.

4. Architectural Trade-Offs and Edge-Native Deployment

The architectural design space for vision-language models in autonomous vehicles can be organized around three interacting axes: backbone efficiency, cross-modal fusion location, and decoding strategy. In early fusion designs, visual and textual representations are combined before deep processing, which can reduce intermediate activation sizes but may sacrifice modality-specific structure. In late fusion designs, separate encoders produce high-level embeddings that are aligned through a lightweight interaction module, which is often more compatible with pre-existing perception stacks. Hybrid designs place cross-attention at selected scales, enabling linguistic context to influence visual interpretation without processing every low-level feature through language-conditioned layers. Each choice affects memory traffic, cache residency, and the feasibility of operator fusion on edge accelerators. Hardware-aware deployment often favors asymmetric designs in which the visual encoder runs on a deterministic neural processing unit and the language decoder runs on a general-purpose processor with flexible control flow.

Pruning, quantization, and distillation are not interchangeable. Unstructured pruning can reduce model size significantly but may produce irregular sparsity that is difficult to exploit on structured accelerators. Structured pruning removes entire channels or attention heads, preserving dense matrix operations and simplifying compilation, but it may remove functional diversity needed for rare scene understanding. Quantization reduces bit width and memory bandwidth, yet its robustness under extreme lighting, adversarial inputs, and sensor degradation must be validated explicitly. Knowledge distillation can help a compact student model approximate the teacher's soft decisions, but the teacher's biases can also be inherited. Hardware-aware optimization should therefore combine these methods in a staged process

that includes sensitivity analysis, robustness evaluation, and continuous monitoring. This process is not a one-time engineering exercise but an ongoing lifecycle practice because automotive fleets operate across changing geographic, climatic, and social conditions.

Edge-native deployment further introduces heterogeneity across vehicle generations and regional hardware ecosystems. Some platforms emphasize high-throughput matrix units, while others optimize for sparse tensor acceleration or low-power digital signal processing. A model optimized for one platform may underutilize another, leading to uneven fleet performance and complicating over-the-air updates. The emergence of edge-native foundation models trained on China-developed accelerators [14] illustrates that platform-specific training can reduce dependency on imported hardware, but it also raises questions about interoperability, standardization, and verification across different automotive markets. From a socio-technical perspective, hardware-aware optimization is therefore entangled with supply chain resilience, technology sovereignty, and regional regulatory divergence. A robust deployment strategy must account for these institutional dimensions rather than assuming a single global hardware target.

5. Runtime Infrastructure and Scheduling for Resource-Constrained Platforms

Runtime infrastructure determines how vision-language inference interacts with conventional perception, planning, and control workloads. Automotive systems increasingly use middleware that manages priority levels, memory partitions, and hardware accelerators under real-time constraints. A vision-language model task may be activated by an unusual-object detector, by remote operator request, or by a passenger query. Such event-triggered workloads are difficult to schedule because their execution time depends on visual token count, language complexity, and decoding length. Middleware must therefore provide admission control, preemption, and graceful degradation. For example, the system may first run a lightweight captioning model to triage an event; only when the preliminary confidence is low may it invoke a larger vision-language reasoner. This layered approach reduces average energy but creates complex coupling between false positives, latency, and task priority inversion.

The energy implications of such scheduling decisions are not neutral. Large-scale neural network training already carries significant carbon and energy costs [15], and the broader policy literature has shown that deep learning workloads in language processing can impose substantial environmental burdens when deployed at scale [16]. Although automotive edge inference is smaller in individual footprint, fleet-level effects become material because millions of vehicles may operate for years. Energy-aware scheduling can dynamically adjust precision, context length, and model capacity according to route complexity, remaining battery, and availability of offloading infrastructure. Such policies must be transparent and auditable, especially when they affect safety-critical semantic understanding in low-power modes.

Infrastructure also includes data logging, model update distribution, and remote supervision. Vision-language models can generate natural-language summaries of unusual scenes that are valuable for fleet incident review and regulatory audits. However, transmitting and storing such summaries raises privacy, consent, and data minimization concerns. Edge-native architectures may keep sensitive visual data on-vehicle and transmit only abstractions, but the linguistic summaries themselves can reveal location, identity, or vulnerable situations. Runtime infrastructure must therefore enforce policy controls such as differential privacy, access logging, and geographic data localization. Hardware-aware optimization affects these controls because smaller models that run locally can reduce the incentive to offload data,

thereby supporting privacy by architecture. The interaction between energy efficiency, latency, and data governance demonstrates that resource-constrained vision-language model deployment cannot be understood as a purely computational problem.

6. Robustness, Redundancy, and Safety Assurance

Deep learning systems have enabled hierarchical feature learning across perception, language, and control tasks [17], but their deployment in safety-critical settings requires explicit treatment of distribution shift, specification ambiguity, and adversarial perturbation. The broader artificial intelligence safety literature identifies problems such as negative side effects, reward hacking, and safe exploration that are highly relevant to autonomous driving even when no reinforcement learning is involved [18]. For vision-language models, robustness failures may be expressed through misleading captions, incorrect visual grounding, or harmful dialogue with passengers. Such failures may not be captured by aggregate accuracy metrics because they are rare, context-dependent, and semantically consequential. Safety assurance therefore demands layered redundancy, where vision-language model outputs are cross-checked against deterministic perception modules, map priors, and physical constraints.

Multi-modal perception in autonomous driving has been studied extensively, with deep architectures fusing camera, lidar, and radar for detection and semantic segmentation [19]. These sensor fusion approaches offer useful lessons for vision-language model robustness because they show how complementary modalities can compensate for individual sensor failures. Similarly, class imbalance and rare-object detection have motivated loss functions such as focal loss to focus learning on hard examples [20]. In a vision-language system, analogous issues arise when rare linguistic queries refer to uncommon objects or unusual traffic situations. Hardware-aware compression may disproportionately degrade performance on rare classes if these classes depend on subtler feature interactions. Robustness evaluation must therefore include adversarial scenes, rare vocabulary, dialectal variation, and degraded visual conditions such as rain, fog, glare, and partial occlusion.

Safety assurance for resource-constrained vision-language models should be organized around risk-based tiers. High-risk functions, such as interpreting traffic control signals or emergency instructions, require higher redundancy, stricter latency bounds, and more conservative model capacity. Lower-risk functions, such as conversational route commentary, can tolerate larger variability and may exploit cloud offloading. This tiering should be enforced at runtime by a safety monitor that evaluates confidence, semantic consistency, and agreement with deterministic perception. The monitor itself must be small, verifiable, and isolated from the learned model. Hardware-aware optimization can support safety by reducing model complexity to a level where formal verification and exhaustive testing become more feasible. However, reducing complexity should not substitute for fail-safe mechanisms. A compressed vision-language model may still produce fluent but incorrect outputs, so the system must treat language generation as an advisory channel rather than an authoritative control signal unless additional validation is present.

7. Sustainability and Energy Governance

Sustainability considerations extend beyond individual vehicles to the entire life cycle of model development, fleet deployment, and hardware manufacturing, maintenance, and eventual decommissioning. The embodied energy of automotive-grade processors, sensors, and memory modules contributes to the total environmental footprint of a vision-language model deployment even before the first kilometer is driven. Moreover, the frequent retraining

cycles that are common in machine learning pipelines can produce substantial carbon emissions unless powered by low-carbon electricity and supported by efficient hardware. Hardware-aware optimization therefore has a direct sustainability role: models that require less compute per inference also reduce the need for high-end silicon, enable longer useful lifetimes for existing vehicle platforms, and lower the frequency of hardware upgrades. This extends the notion of efficiency from per-inference latency to fleet-level resource stewardship.

Energy governance for autonomous mobility involves not only technical efficiency but also accountability for the distribution of environmental costs. A large urban fleet may shift computational load between onboard units and roadside edge nodes, creating new patterns of energy demand that are unevenly distributed across neighborhoods. Policy instruments such as carbon accounting for inference workloads, eco-labeling of model deployment configurations, and public reporting of fleet energy intensity can make these trade-offs visible. Hardware-aware optimization should be conducted with such governance mechanisms in mind, so that compression decisions do not inadvertently externalize energy burdens onto lower-income districts or regions with less resilient electricity grids. The environmental and social dimensions are therefore inseparable from the system design process.

Another governance dimension concerns hardware longevity and circular economy. Overly aggressive optimization for a specific accelerator generation may shorten the effective life of the model when that hardware becomes obsolete, forcing premature vehicle software updates or hardware retrofits. Conversely, conservative optimization that preserves generalism may enable longer deployment but at higher energy cost. Institutional mechanisms such as independent audits of model update cycles, open reporting of hardware compatibility windows, and extended producer responsibility for embedded compute modules can align economic incentives with sustainability goals. Hardware-aware optimization should be framed as a lifecycle decision rather than a single deployment event.

8. Fairness, Accountability, and Sociotechnical Implications

Fairness in vision-language models for autonomous vehicles is complicated by the intersection of algorithmic bias, sensor bias, and spatial inequality. Training data for driving scenes are often overrepresented in affluent urban areas with well-maintained infrastructure, while rural roads, informal settlements, and non-standard signage remain underrepresented. Vision-language models may produce fluent descriptions that nonetheless misclassify pedestrians, street vendors, cyclists, or temporary roadworks in ways that reinforce existing mobility exclusions. The survey literature on bias and fairness in machine learning documents how representational harms can persist even after accuracy improves [21]. Hardware-aware optimization can worsen these harms if pruning and quantization disproportionately affect rare but socially important visual categories. Therefore, fairness evaluation must be integrated into the compression pipeline, with disaggregated metrics across geographic, demographic, and environmental conditions.

Accountability requires that vision-language outputs be traceable and contestable. When a compressed model fails to describe a traffic scene correctly, the cause may lie in the original training distribution, the distillation process, the runtime scheduler, or the sensor degradation. Without systematic logging and provenance tracking, these failures become opaque. Explainable artificial intelligence methods have been proposed to make black-box decisions more interpretable [22], but explanation is not a substitute for responsibility assignment. In automotive contexts, accountability also involves institutional structures such as incident review boards, independent auditing, and regulatory reporting. Hardware-aware optimization

can support accountability by reducing model complexity to a point where causal attribution is more feasible, but it must not be used as an excuse for lower safety standards. The system should maintain an audit trail of model version, compression method, quantization parameters, and runtime configuration.

From a sociotechnical perspective, the deployment of vision-language models in vehicles is not a neutral technical act. It shapes how drivers, passengers, and pedestrians interact with automated systems, and it influences public trust in autonomous mobility. Edge-native deployment may increase privacy by keeping data local, but it also concentrates interpretive power in the vehicle manufacturer. Communities that lack representation in training data may find that the system misinterprets their environment, further marginalizing them. Hardware-aware optimization should therefore be accompanied by participatory design processes, community impact assessments, and clear channels for redress. The choice of model size, precision, and runtime policy has implications for who can audit the system, who bears the cost of failure, and who benefits from efficiency gains. These sociotechnical dimensions are often overlooked in engineering-focused compression research.

9. Regulatory and Policy Frameworks for Edge AI in Mobility

Regulatory frameworks for autonomous vehicles are evolving rapidly, with different jurisdictions emphasizing safety certification, data protection, and algorithmic transparency. The European Union's approach to trustworthy artificial intelligence stresses human oversight, technical robustness, privacy, and accountability [23]. However, hardware-aware model compression introduces specific regulatory challenges. A model that passes certification in its full-precision form may exhibit different failure modes after quantization or pruning, and regulators may not have access to the compression methodology. Certification processes must therefore include requirements for re-validation after any optimization step, as well as ongoing monitoring of field performance. This places new burdens on manufacturers but also creates opportunities for standardized benchmarks that evaluate hardware-aware vision-language models under realistic automotive conditions.

Edge intelligence research has highlighted the benefits of moving computation closer to data sources while also raising concerns about heterogeneous device capabilities and update management [24]. In the automotive domain, over-the-air updates can deliver improved vision-language models, but they also introduce risks of unauthorized changes, version fragmentation, and security vulnerabilities. Regulators may require that critical model updates undergo independent review before deployment, analogous to software updates in aviation. Hardware-aware optimization intersects with these policy questions because smaller models that run efficiently on older hardware may enable longer support lifetimes and reduce forced obsolescence. However, if optimization is proprietary and not externally auditable, public trust may erode. Policy frameworks should encourage open reporting of model compression effects on safety, energy, and fairness while protecting legitimate intellectual property.

International harmonization is an additional challenge. Vehicles are manufactured and sold across borders, and a vision-language model optimized for one region's hardware and data distribution may not meet another region's certification standards. The required reference to an edge-native text-to-image foundation model trained entirely on domestic accelerators [14] illustrates how technology sovereignty and regional hardware ecosystems are becoming increasingly important. This does not necessarily imply fragmentation; it can also motivate interoperable standards for model exchange, performance reporting, and safety assurance across heterogeneous platforms. Hardware-aware optimization can serve as a bridge between

technical efficiency and regulatory legitimacy if it is embedded within transparent governance structures that include regulators, independent experts, and affected communities.

10. Future Directions and Research Agenda

Future research should address the co-design of vision-language architectures and automotive accelerators more explicitly. Current optimization tools often assume a fixed hardware target, but emerging platforms allow reconfigurable dataflows, mixed-precision tensor cores, and in-memory computing. Co-design methods could jointly search for model structures and accelerator configurations that satisfy multi-objective constraints on latency, energy, robustness, and certifiability. This requires moving beyond single-objective benchmarks and developing evaluation suites that capture tail behavior, rare classes, and cross-modal consistency under degraded sensor input. Such benchmarks should be open, versioned, and tied to realistic automotive scenarios.

Another promising direction is the integration of formal verification with learned vision-language components. Although full formal verification of large models remains intractable, abstraction and runtime monitoring can provide bounded guarantees for specific safety properties. Hardware-aware optimization can support this by reducing model complexity and by generating deterministic execution schedules that are easier to analyze. Research on neural network verification for perception systems has advanced significantly, but extending these methods to cross-modal language generation remains an open challenge. A modular architecture that separates visual grounding from language decoding may offer a tractable path toward partial verification.

Finally, the governance of hardware-aware optimization deserves sustained interdisciplinary attention. As autonomous vehicle fleets grow, the aggregate energy, data, and safety implications of model compression will become more visible. Researchers from machine learning, computer architecture, human-computer interaction, law, and environmental science should collaborate to develop standards for reporting compression trade-offs, auditing edge deployments, and assessing community impacts. The articulation of these trade-offs in a transparent and inclusive manner will be essential for the responsible adoption of vision-language models in resource-constrained autonomous vehicles.

11. Conclusion

This paper has argued that hardware-aware optimization of vision-language models for autonomous vehicles is a systemic challenge that extends well beyond model compression. It involves architectural co-design, runtime scheduling, safety assurance, environmental governance, fairness, and regulatory alignment. The structural trade-offs among pruning, quantization, distillation, and backbone selection must be evaluated under realistic constraints that include tail latency, memory bandwidth, thermal budgets, and adversarial robustness. Moreover, these trade-offs are not value-neutral; they affect who benefits from efficient deployment, who bears the risks of failure, and how public accountability is maintained. Edge-native foundation models trained on domestic accelerators [14] demonstrate that hardware-aware optimization is shaped by broader geopolitical and economic forces. The responsible path forward requires cross-layer engineering practices that integrate technical metrics with institutional review, community engagement, and transparent reporting. Only through such a holistic approach can vision-language models contribute to dependable, equitable, and sustainable autonomous mobility.

References

1. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. Advances in Neural Information Processing Systems, 30.

2. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., & Houlsby, N. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929.

3. Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., & Sutskever, I. (2021). Learning transferable visual models from natural language supervision. In International Conference on Machine Learning (pp. 8748-8763). PMLR.

4. Bojarski, M., Del Testa, D., Dworakowski, D., Firner, B., Flepp, B., Goyal, P., Jackel, L. D., Monfort, M., Muller, U., Zhang, J., Zhang, X., Zhao, J., & Zieba, K. (2016). End to end learning for self-driving cars. arXiv preprint arXiv:1604.07316.

5. Grigorescu, S., Trasnea, B., Cocias, T., & Macesanu, G. (2020). A survey of deep learning techniques for autonomous driving. Journal of Field Robotics, 37(3), 362-386.

6. Geiger, A., Lenz, P., & Urtasun, R. (2012). Are we ready for autonomous driving? The KITTI vision benchmark suite. In 2012 IEEE Conference on Computer Vision and Pattern Recognition (pp. 3354-3361). IEEE.

7. Caesar, H., Bankiti, V., Lang, A. H., Vora, S., Liong, V. E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., & Beijbom, O. (2020). nuScenes: A multimodal dataset for autonomous driving. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 11621-11631).

8. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 770-778).

9. Han, S., Mao, H., & Dally, W. J. (2015). Deep compression: Compressing deep neural networks with pruning, trained quantization and Huffman coding. arXiv preprint arXiv:1510.00149.

10. Jacob, B., Kligys, S., Chen, B., Zhu, M., Tang, M., Howard, A., Adam, H., & Kalenichenko, D. (2018). Quantization and training of neural networks for efficient integer-arithmetic-only inference. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 2704-2713).

11. Hinton, G., Vinyals, O., & Dean, J. (2015). Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531.

12. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L. C. (2018). MobileNetV2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 4510-4520).

13. Tan, M., & Le, Q. (2019). EfficientNet: Rethinking model scaling for convolutional neural networks. In International Conference on Machine Learning (pp. 6105-6114). PMLR.

14. Chen, Ce, et al. "JuZhou 1.0 Technical Report: The First Edge-Native Text-to-Image Foundation Model Trained Entirely on China-Developed AI Accelerators." arXiv preprint arXiv:2606.28421 (2026).

15. Strubell, E., Ganesh, A., & McCallum, A. (2019). Energy and policy considerations for deep learning in NLP. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (pp. 3645-3650).

16. Schwartz, R., Dodge, J., Smith, N. A., & Etzioni, O. (2020). Green AI. Communications of the ACM, 63(12), 54-63.

17. LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. Nature, 521(7553), 436-444.

18. Amodei, D., Olah, C., Steinhardt, J., Christiano, P., Schulman, J., & Mané, D. (2016). Concrete problems in AI safety. arXiv preprint arXiv:1606.06565.

19. Feng, D., Haase-Schütz, C., Rosenbaum, L., Hertlein, H., Glaeser, C., Timm, F., Wiesbeck, W., & Dietmayer, K. (2021). Deep multi-modal object detection and semantic segmentation for autonomous driving: Datasets, methods, and challenges. IEEE Transactions on Intelligent Transportation Systems, 22(3), 1341-1360.

20. Lin, T. Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2017). Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision (pp. 2980-2988).

21. Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K., & Galstyan, A. (2021). A survey on bias and fairness in machine learning. ACM Computing Surveys, 54(6), 1-35.

22. Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., & Pedreschi, D. (2018). A survey of methods for explaining black box models. ACM Computing Surveys, 51(5), 1-42.

23. Cath, C. (2018). Governing artificial intelligence: ethical, legal and technical opportunities and challenges. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 376(2133), 20180080.

24. Zhou, Z., Chen, X., Li, E., Zeng, L., Luo, K., & Zhang, J. (2019). Edge intelligence: Paving the last mile of artificial intelligence with edge computing. Proceedings of the IEEE, 107(8), 1738-1762.