Deep Learning and Embedded Systems for Vehicular Traffic Data Analysis

Yosr Ranim Slim[1], Joaquim Ferreira[2], Chokri Souani[3]

1 Laboratory of Electronic and Microelectronic, Faculty of Sciences of Monastir, University of Monastir, Tunisia, Yosrslim24@gmail.com
2 Instituto de Telecomunicaçoes, Departamento de Eletronica, Telecomunicaçoes e Informàtica, Universidade de Aveiro, Portugal, jjcf@ua,pt
3 Higher Institute of Applied Sciences and Technology of Sousse, University of Sousse Sousse, Tunisia, chokri.souani@gmail.com

INTRODUCTION & AIM

Context

The rapid expansion of intelligent transportation systems (ITS) and connected vehicles generates massive volumes of diverse, heterogeneous traffic data — from on-board sensors, LiDAR, cameras, and vehicle-to-everything (V2X) communications.

Problem Sttatement

Efficient classification of this data is critical for mobility, prediction, and safety. While deep learning excels at capturing spatio-temporal dependencies, most approaches remain software-only — overlooking the real constraints of embedded, resource-limited hardware.

Core Gap This Review Addresses

A fundamental disconnect exists between deep learning model design — optimized for accuracy on GPU-equipped servers — and the practical realities of deploying these models in real-time, power-constrained embedded systems such as those found in autonomous and connected vehicles.

Aims of this Review

- Survey DL architectures applied to vehicular traffic data classification
- Evaluate model effectiveness, accuracy, and computational efficiency
- Review optimization strategies enabling deployment on constrained hardware
- Identify open challenges and future directions for real-time embedded AI

METHOD

This work follows a structured literature review methodology, systematically surveying published research at the intersection of deep learning, traffic data classification, and embedded systems deployment.

Inclusion criteria

- DL models applied to vehicular or traffic data
- Peer-reviewed, published 2015–2024
- Reports model performance
- Discusses hardware deployment

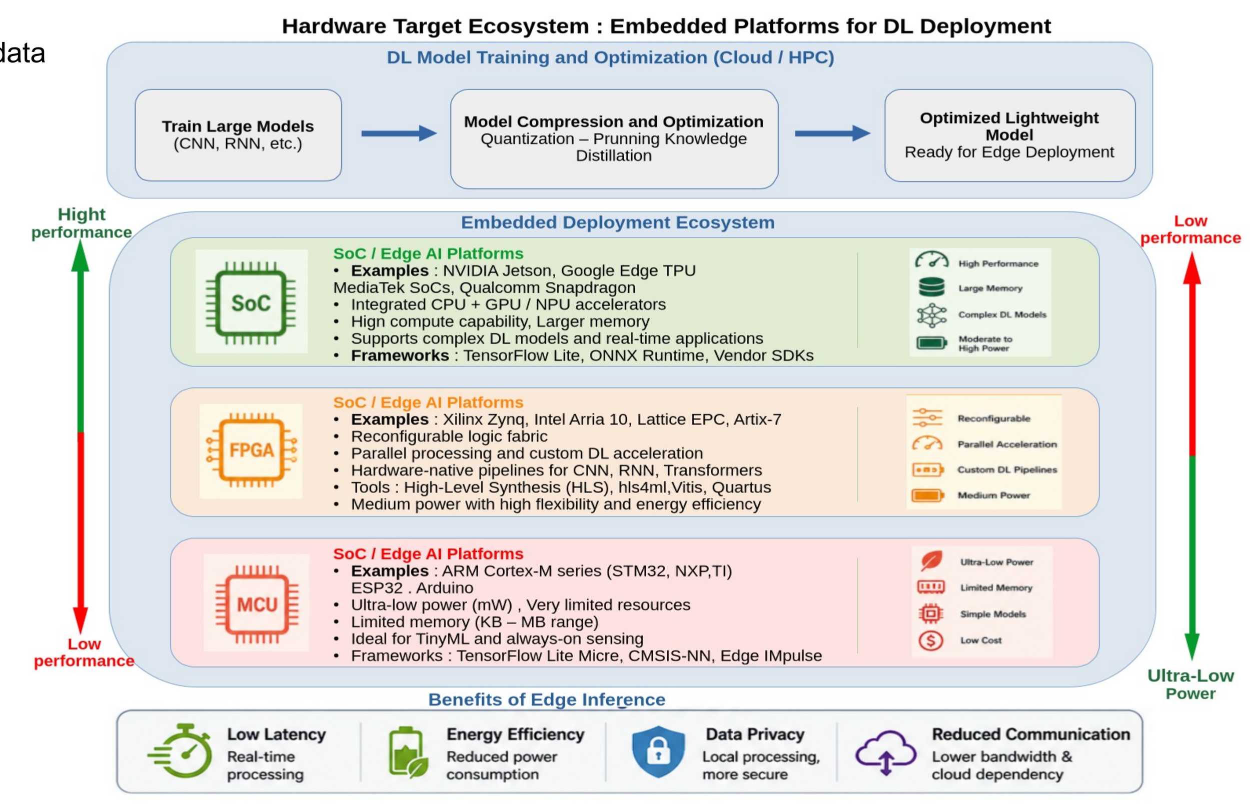
Analytical dimensions

- Model architecture taxonomy :
- CNN · RNN · GCN · Transformer
- Performance benchmarking :
- accuracy · F1 · MAE
- Optimization strategies :
- quant. · pruning · distil. · NAS
- Hardware deployment context :
- MCU · FPGA · SoC · edge

Keywords used in literature search

Deep learning, Traffic data classification, Intelligent transportation systems, Embedded systems, Model compression, Real-time inference, Connected vehicles, Spatio-temporal modeling, Edge AI, TinyML, Quantization, Autonomous vehicles

Hardware Target Ecosystem : Embedded Platforms for DL Deployment



RESULTS & DISCUSSION

Deep Learning Outperforms Traditional Methods

Deep learning architectures consistently outperform traditional machine learning methods, on traffic data classification tasks. Spatio-temporal models that jointly model both spatial topology and temporal dynamics achieve the strongest results.
CNN extract local spatial patterns from traffic maps and sensor grids; LSTM/GRU capture sequential temporal dependencies; GCN model road network topology; Transformers extend this with long-range attention.

Deep Learning (Spatio-Temporal Models)
CNN LSTM/GRU GCN TRANSFORMER
Consistently Higher Performance

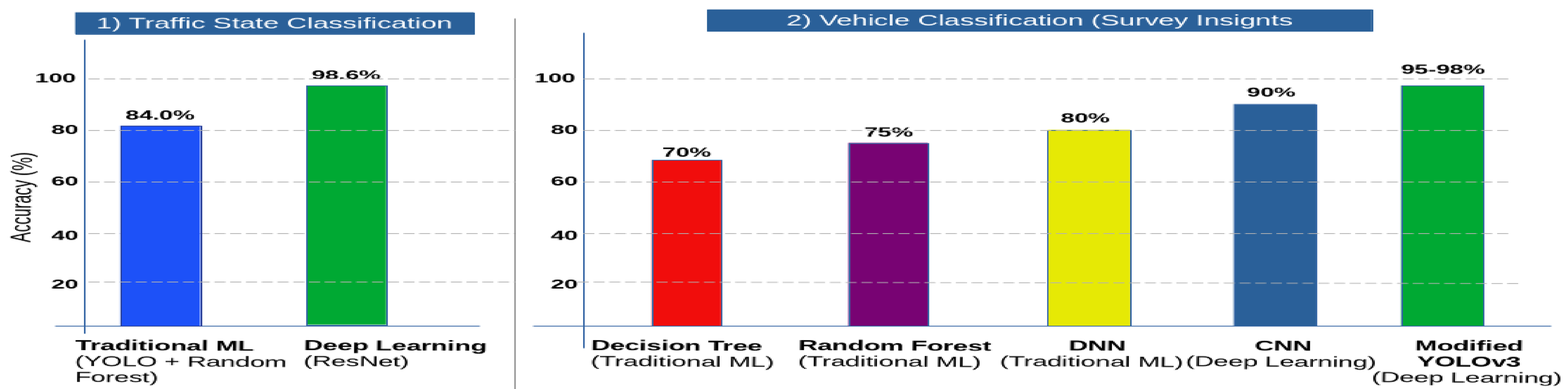
Traditional Machine Learning
SVM Random Forest ARIMA
Lower Performance

CNN
Extracts local spatial patterns from trafic maps and sensor grids.

LSTM
Captures sequential temporal dependencies in trafic flows.

GCN
Models road network topology and learns from neighboring nodes.

Transformer
Captures long-range dependencies with attention across all positions.



Deep Learning (98.6 %) outperforms Traditional ML (84 %) by ~14.6 percentage points.

Deep Learning models (CNN, YOLOv3) achieve the highest accuracy, significantly outperforming traditional ML approaches.

Traditional ML Approaches (Hand-crafted features + Classic Classifiers)

Deep Learning Approaches (Learned Features + Deep Architectures)

Note: Values are based on reported results from the following papers: Impedovo, D.; Balducci, F.; Dentamaro, V.; Pirlo, G. Vehicular Traffic Congestion Classification by Visual Features and Deep Learning Approaches: A Comparison. Sensors 2019, 19, 5213. https://doi.org/10.3390/s19235213. Berwo, M.A.; Khan, A.; Fang, Y.; Fahim, H.; Javaid, S.; Mahmood, J.; Abideen, Z.U.; M.S., S. Deep Learning Techniques for Vehicle Detection and Classification from Images/Videos: A Survey. Sensors 2023, 23, 4832. https://doi.org/10.3390/s23104832

Optimization Strategies Enable Embedded Deployment

Model optimization is critical for deploying deep learning on resource-constrained embedded platforms.
These strategies significantly reduce model size, memory footprint, and inference energy while maintaining accuracy.

1. Quantization (FP32-->INT8)
Reduces model size up to **4x** and inference energy with minimal accuracy loss ; essential for MCU deployment.

2. Structured Pruning
Removes redundant filters and neurons ; can compress networks up to **13x** while preserving classification accuracy.

3. Knowledge Distillation
Teacher-student training produces compact student models that retain most of the teacher's predictive capacity.

4. Neural Architecture Search (NAS)
Automates hardware-aware design, producing architectures tailored to specific MCU or FPGA memory and compute budgets.

These optimization strategies make deep learning models practical for real-world embedded systems with strict memory, compute, and energy constraints.

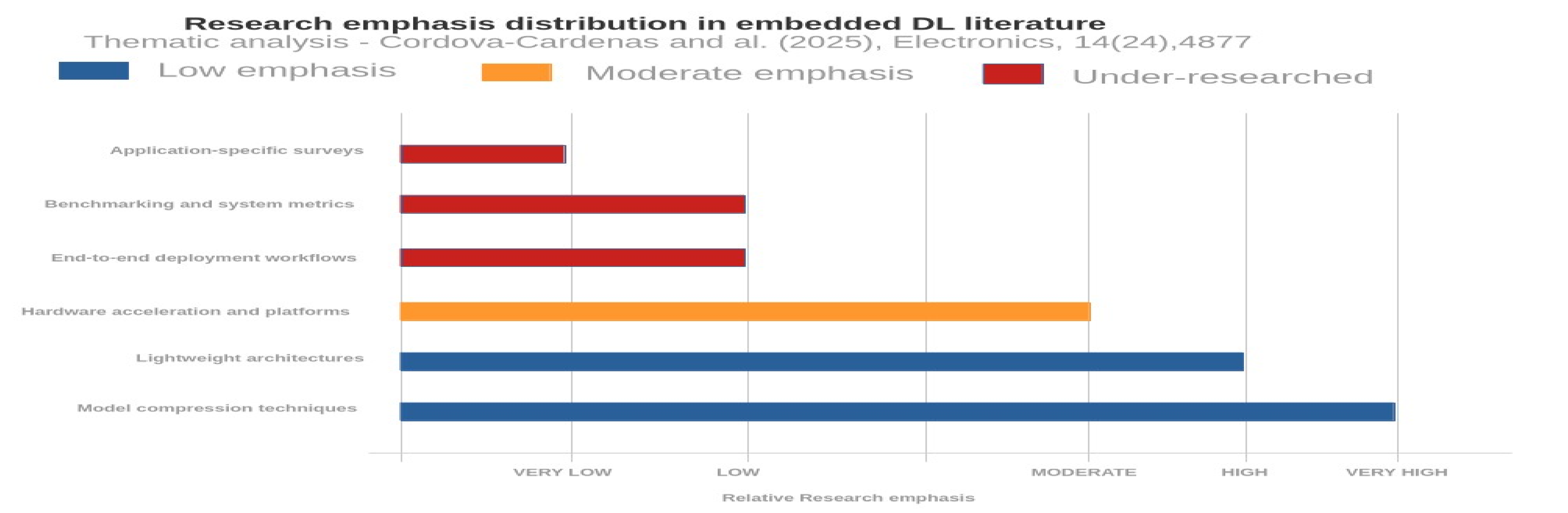
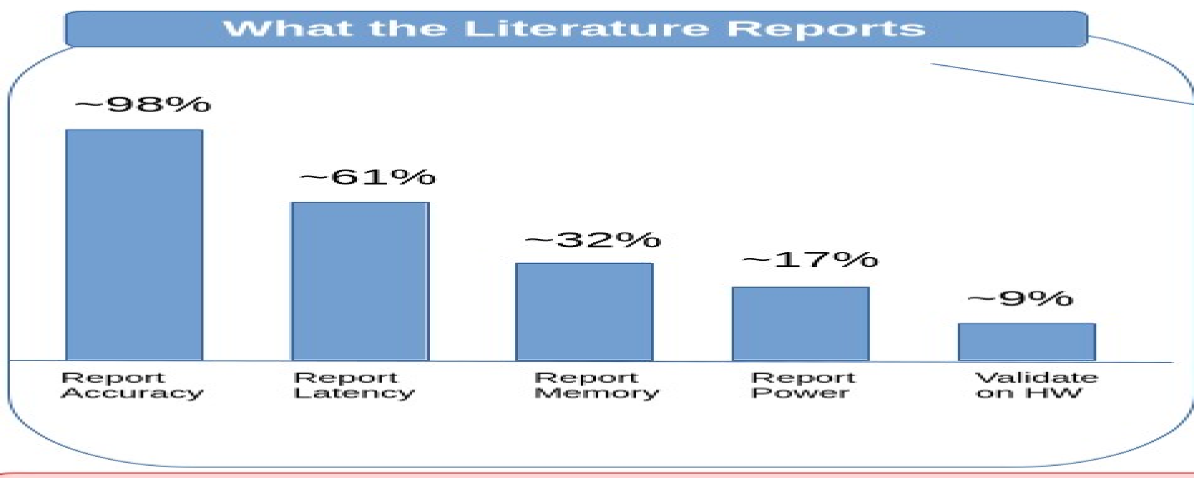


Figure 3 : Relative research emphasis across six thematic areas in the embedded deep learning literature, based on thematic analysis of Cordova-Cardenas et al.

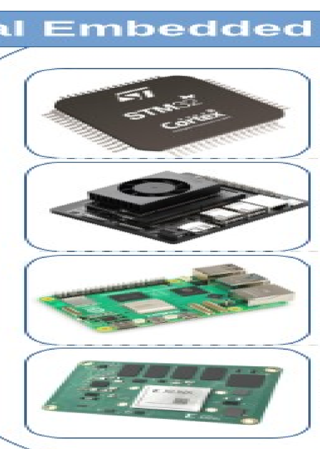
A Critical Deployment Gap Exists in the Literature

The vast majority of reviewed studies evaluate DL models exclusively on GPU-equipped servers, reporting accuracy metrics without profiling latency, memory footprint, or power consumption on real embedded hardware. Hardware validation on platforms such us STM32, NVIDIA Jetson, Raspberry Pi, or Xilinx FPGA remains rare. This constitues a major gap between published results and real-world deployability.

What the Literature Reports



Real Embedded Hardware Validation Remains Rare



STM32 Microcontroller platforms (e.g., STM32F4, STM32H7)

NVIDIA Jetson Edge AI computers (e.g., Nano, Xavier, Orin)

Raspberry Pi Low cost single board computers (e.g., Pi 4, Pi 5)

Xilinx FPGA Programmable SoCs (e.g., Zynq, UltraScale+)

!! This gap between evaluation and real-world deployment limits the reliability and reproducibility of published results for ressource-constrained embedded applications.

CONCLUSIONS

Deep learning dominates traffic data classification, outperforming traditional methods, especially with hybrid spatio-temporal models. However, deploying these models on embedded systems remains a major challenge. Despite advances in optimization techniques, real hardware validation is still largely missing, creating a gap between high performance and practical deployment.

DL wins
Consistently outperforms traditional ML on all traffic tasks

Compression
Quantization and pruning make embedded deployment feasible

Gap exists
Software-only evaluation dominates; HW validation is rare

Co-design
Hardware-aware model design is the critical path forward

FUTURE WORK/ REFERENCES

FUTURE WORK

- Hardware-aware co-design
- Standardised embedded benchmarks
- Federated & on-device learning
- TinyML & neuromorphic computing
- Multi-modal sensor fusion at the edge

REFERENCES

- Berwo, M.A.; et al. Deep learning techniques for vehicle detection and classification from images/videos: A survey. Sensors 2023, 23, 4832.
- Impedovo, D.; et al. Vehicular traffic congestion classification by visual features and deep learning approaches: A comparison. Sensors 2019, 19, 5213.
- Cordova-Cardenas, R.; Amor, D.; Gutiérrez, Á. Edge AI in practice: A survey and deployment framework for neural networks on embedded systems. Electronics 2025, 14, 4877.
- Alajlan, N.N.; Ibrahim, D.M. TinyML: Enabling inference of deep learning models on ultra-low-power IoT edge devices for AI applications. Micromachines 2022, 13, 851.