

FPGA-based MLP Hardware Accelerator for Real-Time Destination Prediction in Autonomous Vehicles

Hyeon Muk Park, Won Seok Choi, Seong Gon Choi*

The Depart of Information and Computer Engineering, Chungbuk National University
Chungdae-ro, Seowon-gu, Cheongju-city, Chungcheongbuk-do, 28644 Republic of Korea
qga0809@chungbuk.ac.kr , choiws@cbnu.ac.kr , choisg@cbnu.ac.kr

Abstract— We propose FPGA-based MLP (Multi-Layer Perceptron) hardware accelerator designed for real-time destination prediction in autonomous vehicles. As personalized destination prediction services in autonomous vehicle systems become increasingly sophisticated, there is a growing need to perform efficient and low-latency AI inference directly within vehicles rather than relying on cloud servers. However, software-based neural network inference on embedded processors creates performance bottlenecks that fail to meet the real-time requirements of autonomous vehicles. The proposed system implements MLP model that takes seven features as input—including Device ID, Device Type, latitude, longitude, time information—and predicts six destination classes. Developed using Vitis HLS on the Zynq-7000 SoC platform, the FPGA hardware accelerator achieves an inference time of 0.005454ms through AXI Stream interface, demonstrating a 68.53× performance improvement compared to ARM-based software implementation (0.373770ms).

Keywords— Destination prediction, FPGA hardware accelerator, MLP neural network, Autonomous vehicles, Zynq SoC

I. INTRODUCTION

As autonomous vehicle technology advances, there is a significantly growing need for intelligent destination prediction services that go beyond simple route guidance to reflect individual preferences and driving patterns. These personalized services not only enhance user convenience by predicting destinations based on historical data and contextual information, but are also gaining attention as key technologies that improve the overall efficiency of autonomous driving systems [1].

For such user-customized AI services to have practical value in automotive environments, real-time responsiveness is essential. Modern vehicles require immediate inference results to provide seamless user experiences and support time-critical decision-making processes. However, software-based neural network inference on embedded processors in vehicles creates significant performance bottlenecks that fail to meet the stringent real-time requirements. The computational intensity of AI algorithms, combined with the limited processing power

of general-purpose processors in automotive systems, results in unacceptable latency for real-time applications [2].

Location-based services such as destination prediction represent a typical field of time-series data analysis. To learn the temporal dependencies of such time-series data, recurrent neural network models such as LSTM (Long Short-Term Memory) and RNN (Recurrent Neural Network) have been widely used in previous research. Earlier studies have focused on achieving high prediction accuracy through LSTM-based software models [4][5]. However, LSTM models have complex gate structures and recurrent data dependencies, making large-scale parallel processing difficult when implemented in hardware and requiring high memory bandwidth. Consequently, this leads to excessive hardware resource consumption, presenting clear limitations for automotive applications where computational efficiency is crucial.

Therefore, We proposes a hardware accelerator based on MLP (Multi-Layer Perceptron) model optimized for automotive embedded environments, replacing the conventional complex LSTM models. The proposed system takes seven features as input: Devid, Devtype, Latitude, Longitude, Devtime, Day, and date_ord. Through performance comparison between Processing System (PS) and Programmable Logic (PL) implementations on the Zynq-7000 SoC platform, we demonstrate the superiority of FPGA hardware acceleration and achieve significant performance improvements compared to ARM-based software implementation.

The organization of this paper is as follows. Section 2 reviews related work, and Section 3 describes the proposed system architecture in detail. Section 4 presents the implementation and verification of the proposed hardware, while Section 5 provides performance analysis and results. Finally, Section 6 discusses conclusions and future research directions.

II. RELATED WORK

In this section, we review recent research efforts in destination prediction for autonomous vehicles and FPGA-

based neural network acceleration technologies that are directly relevant to our proposed approach.

A. Destination Prediction in Autonomous Vehicles

Destination prediction in automotive environments has emerged as a critical component for enhancing user experience and optimizing autonomous driving systems. The primary goal is to predict future destinations based on historical movement patterns, temporal information, and contextual data such as device types and user behaviors.

Recent studies have demonstrated significant achievements in destination prediction using deep learning approaches, particularly LSTM-based models. Choi et al. [4] implemented an LSTM-based framework to predict future locations using historical location data, focusing on learning temporal dependencies from past movement patterns. Jeong et al. [5] developed an LSTM-based destination prediction method that incorporates both location information and temporal characteristics such as time-of-day and day-of-week patterns to enhance prediction accuracy.

However, these studies have primarily focused on improving prediction accuracy through software-based implementations without considering hardware deployment aspects [4][5]. While these LSTM-based approaches demonstrate promising results in software environments, they do not address the hardware implementation challenges that are crucial for practical deployment in automotive embedded systems.

Therefore, there is a clear need for destination prediction approaches that are designed with hardware efficiency in mind, particularly for real-time deployment in resource-constrained automotive environments where hardware acceleration becomes essential.

B. FPGA-based Neural Network Acceleration

Field-Programmable Gate Arrays (FPGAs) have emerged as a promising platform for accelerating neural network inference due to their reconfigurable architecture and excellent parallel processing capabilities. The ability to customize hardware architectures for specific computational patterns makes FPGAs particularly attractive for deploying neural networks in resource-constrained environments.

Several studies have explored FPGA acceleration of LSTM networks for sequential data processing applications. Han et al. [6] developed ESE (Efficient Speech Recognition Engine), an FPGA-based LSTM accelerator that achieved significant speedups through sparse matrix operations and specialized hardware design. However, their implementation revealed that LSTM's inherent sequential dependencies and complex gate structures present fundamental challenges for FPGA deployment. Zhang et al. [7] implemented an LSTM accelerator using pipeline optimization techniques. However, despite these optimization efforts, the recurrent nature of LSTM computation inherently limits parallelization opportunities and requires extensive memory bandwidth for state management.

The computational complexity of LSTM stems from its three-gate architecture (forget, input, and output gates) and

sequential processing requirements. Wang et al. [8] demonstrated that even with structured pruning techniques, LSTM implementations on FPGAs consume substantial hardware resources due to the need for complex control logic and extensive BRAM utilization for storing intermediate states. The sequential data dependencies inherent in LSTM architectures prevent effective parallelization, resulting in suboptimal hardware utilization and increased latency.

In contrast, Multi-Layer Perceptron (MLP) architectures offer significant advantages for FPGA implementation. The feed-forward structure of MLPs enables straightforward parallelization without the sequential dependencies that constrain LSTM implementations [9]. Recent studies have shown that MLP accelerators can achieve higher throughput and lower latency compared to recurrent architectures while requiring fewer hardware resources [10]. The matrix multiplication operations in MLPs can be efficiently mapped to systolic array structures on FPGAs through optimized systolic array implementations, enabling high-performance parallel computation with predictable timing characteristics [11].

Building upon these findings, our work leverages the hardware-friendly characteristics of MLP architectures to achieve real-time destination prediction performance suitable for autonomous vehicle applications.

III. PROPOSED METHOD

This section presents our FPGA-based MLP hardware accelerator designed for real-time destination prediction in autonomous vehicles. The proposed approach consists of a three-tier system architecture that separates data collection, model training, and inference acceleration to achieve optimal performance while maintaining data privacy.

A. System Architecture Overview

Figure 1 illustrates the overall system architecture of our proposed destination prediction system. The architecture employs a three-tier approach that separates data collection, model training, and inference execution to optimize performance and maintain data privacy.

The OBU (On-Board Unit) serves as the primary interface within the vehicle environment. It encompasses three core functionalities: data collection from various sensors and contextual sources, data transmission capabilities for communication with external systems, and a dashboard interface for real-time monitoring and result visualization. The OBU continuously gathers contextual information including device identification, location coordinates, temporal data, and user interaction patterns.

The Private Server functions as the centralized machine learning hub responsible for model training and management. It receives training data from multiple OBU sources, stores the aggregated information securely, and performs model training using the collected datasets. The server maintains data privacy by processing information locally rather than relying on external cloud services.

The ZYNQ Platform operates as the dedicated inference engine that executes real-time destination predictions. It receives trained model parameters from the Private Server and processes current context data from the OBU to generate immediate prediction results. This architecture enables low-latency inference by performing computations locally without requiring network communication during the prediction phase.

The operational workflow is partitioned into two distinct phases: model training and real-time inference. In the training phase, historical contextual data is gathered by the OBU ("Data Collect") and transmitted ("Data Transmit") to the Private Server. The server aggregates this data ("Data Receive," "Data Store") and utilizes it to train the MLP model ("Model Train"). The resulting model parameters ("Trained Data") are then deployed to the "Destination Inference Accelerator" on the ZYNQ platform. Subsequently, during the real-time inference phase, the OBU bypasses the server and sends "Current Context Data" directly to the ZYNQ's "Data Receive" module. The accelerator processes this input using the deployed parameters and sends the "Prediction Results" back ("Result Transmit") to the OBU's "Dashboard" for visualization.

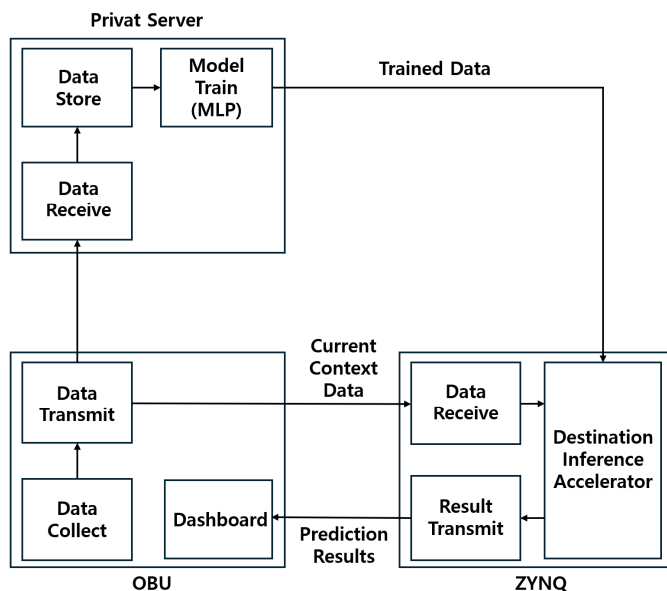


Figure 1. Overall System Architecture

B. MLP Model Design

Figure 2 depicts the detailed structure of our proposed MLP model, which has been specifically designed for efficient FPGA implementation. The network follows a four-layer feed-forward architecture that balances prediction accuracy with hardware resource constraints.

The Input Layer accepts seven features that capture the essential contextual information for destination prediction. These features represent both spatial (latitude, longitude) and temporal (timestamp, day, date_ord) characteristics, along with device-specific information (device ID, device type) that enables personalized predictions.

The Hidden Layers employ a gradually decreasing neuron configuration: Hidden Layer 1 with 64 neurons, Hidden Layer 2 with 32 neurons, and Hidden Layer 3 with 16 neurons. This pyramidal structure enables the network to progressively extract higher-level features while reducing computational complexity for subsequent layers. Each hidden layer utilizes ReLU activation functions to introduce non-linearity while maintaining computational efficiency for FPGA implementation.

The Output Layer implements a softmax activation function to produce probability distributions across six destination classes. This classification approach enables the system to predict the most likely destination category based on the current contextual input, providing actionable information for autonomous vehicle navigation systems.

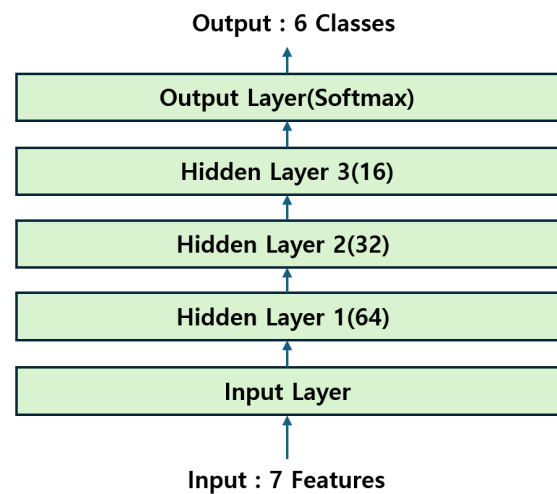


Figure 2. Proposed MLP Network Architecture for Destination Prediction

IV. IMPLEMENTATION

This section details the hardware implementation of our proposed MLP-based destination prediction system on the FPGA platform, including the development environment, hardware optimization techniques, and experimental validation.

Our implementation is built on the Zynq-7000 SoC platform, specifically targeting the ZedBoard development platform. The system leverages the dual-core ARM Cortex-A9 processor in the Processing System (PS) for system control and communication, while the Programmable Logic (PL) hosts the optimized MLP hardware accelerator.

The hardware accelerator is developed using Vitis HLS 2022.2, which enables high-level synthesis from C++ descriptions while providing fine-grained control over hardware optimization. Vivado 2022.2 is used for place-and-route implementation and bitstream generation. The AXI Stream interface protocol facilitates efficient data transfer between the PS and PL components, ensuring minimal communication overhead during real-time inference operations.

```

=== PL Hardware Results ===
Raw Logits:
Home1: 8.201172
Home2: 3.445312
Gym: -3.929688
Hospital: -10.327148
Yeoungooong: -5.355469
CBNU: 1.022461

Probabilities:
Home1: 0.9907 (99.1%)
Home2: 0.0085 (0.9%)
Gym: 0.0000 (0.0%)
Hospital: 0.0000 (0.0%)
Yeoungooong: 0.0000 (0.0%)
CBNU: 0.0008 (0.1%)

*** Top Prediction: Home1 (99.1%) ***

```

Figure 3. Inference Output with Destination Probabilities

```

=====
Performance Analysis (Average)
=====
*** Average PS Software Inference Time: 0.373770 ms (over 1000 runs)
*** Average PL Hardware Inference Time: 0.005454 ms (over 1000 runs)
*** Speedup (PL vs PS): 68.53 x ***

```

Figure 4. Performance Analysis Results

Figure 3 illustrates a representative inference result where the FPGA accelerator successfully processes input context data and generates classification probabilities. In this specific test case, the system predicts "Home1" as the most likely destination with 99.1% confidence, while other destinations receive significantly lower probabilities: Home2 (0.9%), CBNU (0.1%), and Hospital, Gym, Yeoungooong (0.0%).

Figure 4 presents the performance analysis results, revealing that the inference process achieves an execution time of 0.005454 ms, representing a 68.53× performance improvement compared to the ARM-based software implementation (0.373770 ms). This significant acceleration demonstrated in the performance metrics enables real-time destination prediction suitable for autonomous vehicle applications requiring sub-millisecond response times.

The experimental results validate the effectiveness of our MLP-based approach in achieving both high prediction accuracy and real-time performance requirements. The clear classification results with high confidence levels demonstrate the model's capability to make reliable destination predictions based on contextual input features.

V. CONCLUSIONS

This paper presented an efficient MLP-based FPGA hardware accelerator for real-time destination prediction in autonomous vehicles. The proposed four-layer feed-forward architecture (7→64→32→16→6) successfully addressed the computational complexity limitations of LSTM-based approaches while achieving significant performance improvements for real-time applications.

Experimental results demonstrate that the proposed system successfully predicts destination classes with high confidence levels while maintaining sub-millisecond response times essential for autonomous vehicle applications. The FPGA implementation effectively utilizes hardware resources

through pipelined architecture and optimized data flow, proving the viability of edge-based AI inference for automotive environments.

Future work will focus on expanding the feature set to incorporate additional contextual information such as traffic conditions and weather data for enhanced prediction accuracy. Furthermore, we plan to investigate multi-vehicle collaborative learning approaches and explore advanced FPGA optimization techniques to support larger neural network architectures while maintaining real-time performance constraints.

ACKNOWLEDGMENT

This work was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (No. RS-2020-NR049604, 100%).

*Corresponding author is S.G. Choi(choisg@cbnu.ac.kr).

REFERENCES

- [1] L. Thomas, S. Hirt, A. Staehle, and R. Findeisen, "Exploring Prior Knowledge in Preferential Learning of Individualized Autonomous Vehicle Driving Styles," in *Proc. 2025 European Control Conference (ECC)*, Thessaloniki, Greece, 2025, pp. 2217-2222, doi: 10.2391/ECC60501.2025.11108687.
- [2] M. L. M. Peixoto, "Quantum Edge Computing for Data Analysis in Connected Autonomous Vehicles," in *Proc. 2024 IEEE Symposium on Computers and Communications (ISCC)*, Paris, France, 2024, pp. 1-6, doi: 10.1109/ISCC61873.2024.10735687.
- [3] J. S. Fara, W. M. Elmouelhi, and K. Elleithy, "Balancing Performance and Cost—FPGA-Based CNN Acceleration for Edge Computing: Status Quo, Key Challenges, and Prospective Innovations," *IEEE Access*, vol. 13, pp. 142852-142877, 2025, doi: 10.1109/ACCESS.2025.3504348.
- [4] W. S. Choi, J. H. Jeong, and S. G. Choi, "LSTM-based Location Prediction using Historical Location Data," in *Proc. KICS Summer Conference*, Jeju, Korea, Jun. 2025.
- [5] J. H. Jeong, W. S. Choi, and S. G. Choi, "LSTM-Based Destination Prediction Method Using Location Information and Temporal Characteristics," in *Proc. Int. Conf. on ICT Convergence (ICTC)*, Jeju, Korea, Oct. 2025.
- [6] S. Han *et al.*, "ESE: Efficient speech recognition engine with sparse LSTM on FPGA," in *Proc. ACM/SIGDA Int. Symp. Field-Programmable Gate Arrays (FPGA)*, Monterey, CA, USA, Feb. 2017, pp. 75-84, doi: 10.1145/3020078.3021741.
- [7] Y. Zhang *et al.*, "Implementation and optimization of the accelerator based on FPGA hardware for LSTM network," in *Proc. IEEE Int. Symp. Parallel and Distributed Process. Appl. and IEEE Int. Conf. Ubiquitous Comput. and Commun. (ISPA/IUCC)*, Guangzhou, China, Dec. 2017, pp. 614-619, doi: 10.1109/ISPA/IUCC.2017.00099.
- [8] S. Wang *et al.*, "Acceleration of LSTM with structured pruning method on FPGA," *IEEE Access*, vol. 7, pp. 62930-62937, 2019, doi: 10.1109/ACCESS.2019.2917300.
- [9] I. Westby and X. Yang, "FPGA acceleration on a multi-layer perceptron neural network for digit recognition," *J. Supercomput.*, vol. 77, no. 5, pp. 4753-4770, May 2021, doi: 10.1007/s11227-020-03437-z.
- [10] F. Li, T. Zhang, B. Yu, and H. Li, "FPGA-QNN: Quantized neural network hardware accelerator for multi-layer perceptron," *Appl. Sci.*, vol. 15, no. 2, p. 688, Jan. 2025, doi: 10.3390/app15020688.
- [11] J. Zhang *et al.*, "Frequency improvement of systolic array-based CNNs on FPGAs," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, Sapporo, Japan, May 2019, pp. 1-4, doi: 10.1109/ISCAS.2019.8702334.



Hyeon Muk Park received B.S. degree in college of Information & Communication Engineering from Chungbuk National University in 2024. He is currently pursuing the M.S. degree in Radio Communication Engineering, Chungbuk National University. His research interests include FPGA, V2X, IoT and AI.



Won Seok Choi received B.S. and Ph.D. degree in the College of Electrical and Computer Engineering, Chungbuk National University, Korea in 2008 and 2014 respectively. He is currently researcher in Research institute of Computer and Information Communication, Chungbuk National University. His research interests include vehicle network, energy saving network, SDN, NFV, NGN and AI.



Seong Gon Choi received B.S. degree in Electronics Engineering from Kyungpook National University in 1990, and M.S. and Ph.D. degree from KAIST in Korea in 1999 and 2004, respectively. He is currently a professor in College of Electrical & Computer Engineering, Chungbuk National University. His research interests include V2X, AI, smart grid, IoT, mobile communication, high-speed network architecture and protocol.