FPGA BASED ML EDGE

Vishwanath Singh
Center for Embedded and Cyber-Physical Systems
University of California Irvine
Irvine, CA, USA
vishws1@uci.edu

Manish Sudumbrekar
Center for Embedded and Cyber-Physical Systems
University of California Irvine
Irvine, CA, USA
msudumbr@uci.edu

***Abstract*— Lane detection is a critical component of advanced driver-assistance systems (ADAS), enabling functionalities such as Lane Keeping Assist and Lane Departure Warning, which contribute to safer driving experiences. This project explores the deployment of lane detection models on the Xilinx Kria KV260 FPGA to enhance real-time performance and energy efficiency. Leveraging the Vitis AI framework, three models—VPGNet, Ultrafast, and YOLOv3—were evaluated on the TuSimple dataset, demonstrating the FPGA's capability to deliver high accuracy and low latency in lane detection tasks. Among the models, VPGNet exhibited superior FPS performance, making it suitable for real-time applications, while Ultrafast achieved the highest accuracy. Future work aims to expand the framework to integrate additional vision tasks, develop adaptive algorithms for complex road conditions, and explore end-to-end AI solutions directly on FPGA hardware. This study underscores the advantages of FPGA acceleration in vision-based systems, particularly in safety-critical applications.**

Keywords—Lane Detection, FPGA, ADAS, Real-Time Processing, Xilinx Kria KV260, Vitis AI, VPGNet, Ultrafast, YOLOv3, TuSimple Dataset, Low Latency, Energy Efficiency, Edge AI, Vision Applications, Safety-Critical Systems.

I. INTRODUCTION

Lane detection is a fundamental task in computer vision, designed to identify and track the boundaries of driving lanes in real-time, even under challenging conditions such as varying lighting, glare, or complex road layouts. This capability plays a pivotal role in advanced driver-assistance systems (ADAS), enabling features like Lane Keeping Assist and Lane Departure Warning, which are proven to reduce accidents and improve road safety. Traditionally, lane detection has relied on CPUs or GPUs for processing, but these approaches face limitations in latency and energy efficiency, particularly in safety-critical applications. This project addresses these challenges by leveraging FPGA-based hardware, specifically the Xilinx Kria KV260, to optimize lane detection models for real-time performance, low latency, and energy efficiency, ensuring robust functionality in diverse road conditions.

II. EASE OF USE

1. Enhance Lane Detection Performance

Real-time lane detection is essential for applications like autonomous vehicles and ADAS. The project aims to leverage FPGA's parallel processing capabilities to deliver high frame rates (FPS) and accurate lane detection in real-world scenarios. By minimizing computational bottlenecks, the system ensures swift detection and response times.

2. Improve Energy Efficiency

Energy efficiency is critical for embedded systems in vehicles. Compared to traditional GPUs, the Xilinx Kria KV260 FPGA consumes significantly less power while maintaining competitive performance. This reduces overall system energy demands, enhancing suitability for battery-powered and energy-sensitive applications.

3. Deploy Advanced Models

The project utilizes cutting-edge lane detection models—VPGNet for robustness, Ultrafast for high accuracy, and YOLOv3 for simplicity—to identify the best fit for FPGA-based deployment. This enables a comparative analysis of model architectures under real-world constraints.

4. Optimize Model Processing

Using the Vitis AI framework, models are optimized through quantization (reducing bit precision to INT8), pruning (removing redundant network layers), and efficient compilation for FPGA hardware. These techniques ensure reduced memory usage and faster execution while maintaining model accuracy.

5. Validate on Real-World Data

The TuSimple dataset, comprising 6,408 high-resolution images captured under diverse weather conditions, provides a robust testing ground. Training and validation across varying scenarios (e.g., rain, snow, and night-time driving) ensure the system's adaptability and reliability.

6. Benchmark Against Other Systems

A performance comparison with NVIDIA's Jetson Nano reveals FPGA's advantages in latency, FPS, and energy efficiency. Metrics like accuracy and robustness under adverse conditions highlight FPGA's superior capability for real-time vision tasks.

7. Adapt to Safety-Critical Systems

Safety-critical systems require high reliability and minimal failure rates. The FPGA implementation is designed to handle challenging conditions such as low light, glare, and complex road layouts with consistent accuracy and robust performance.

8. Integrate with ADAS Features

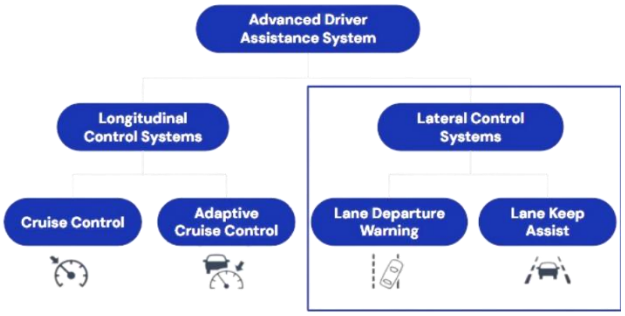

Figure 1: Sub systems of advance diver assistance system

Lane detection is a cornerstone of ADAS functionalities. This project integrates lane detection with existing systems like Lane Keeping Assist and Lane Departure Warning to enhance vehicle safety, providing real-time alerts or corrective actions.

9. Promote Scalability

The developed framework is not limited to lane detection. It is scalable to integrate other edge AI applications, such as pedestrian tracking, traffic sign recognition, and object detection, making it versatile for broader automotive and non-automotive use cases.

10. Explore Future Innovations

The project lays the groundwork for future enhancements, including lightweight AI models natively trained on FPGA hardware for reduced overhead and greater efficiency. Additionally, dynamic adaptation to evolving road conditions, such as merging lanes or bad weather, is proposed to improve usability and reliability further.

III. Hardware and Software Setup

The hardware foundation for the project was the Xilinx Kria KV260 Vision AI Starter Kit, chosen for its capabilities tailored to vision applications. This FPGA board features an ARM Cortex-A53 4-core processor, 4 GB DDR memory, 512 MB primary boot memory, and 1 Gb Ethernet connectivity, making it ideal for handling high-performance lane detection tasks. The development environment included Linux-based systems and the Vitis AI framework, which provides tools for optimizing, quantizing, and deploying neural network models onto FPGA hardware. Supporting software like Vivado and runtime libraries ensured efficient execution and seamless integration of the models.

Figure 2: Xilinx Kria Kv260 Zynq™ UltraScale+™ MPSoC

Three state-of-the-art lane detection models were implemented: VPGNet for robustness in diverse conditions, Ultrafast (ResNet18) for maximum accuracy, and YOLOv3 for a simplified yet effective approach. These models were trained and validated using the TuSimple dataset, comprising 6,408 high-resolution images (1280×720) under various weather conditions, including rain, snow, and night-time driving. The dataset provided a realistic testing ground for evaluating the models' performance in real-world scenarios.

Optimization of the models leveraged the Vitis AI toolkit, which facilitated the quantization of 32-bit floating-point weights to INT8, significantly improving execution speed while maintaining accuracy. Pruning redundant kernels further reduced computational load. Once optimized, the models were compiled using the Vitis AI Compiler, mapping them to the FPGA's instruction set, and then deployed using runtime API functions for seamless operation.

The performance metrics focused on frames per second (FPS), power consumption (ranging from 7.5 to 15 W), and accuracy. To highlight the FPGA's advantages, the setup was benchmarked against NVIDIA's Jetson Nano, revealing superior latency, energy efficiency, and real-time processing capabilities for the Xilinx board. Rigorous testing was conducted in various environments, simulating challenging weather conditions using the TuSimple dataset to ensure robustness.

IV. Standards Used for Projects

The project follows established standards to ensure robustness, scalability, and compatibility. Lane detection accuracy is evaluated using standard metrics like Intersection over Union (IoU) and frame accuracy, while FPGA design adheres to the Xilinx Vitis AI framework and IEEE hardware description language standards. Neural network optimization follows industry best practices, including quantization and pruning, ensuring compatibility with edge hardware. The TuSimple dataset, a widely recognized benchmark for lane detection, was used to maintain consistency with industry evaluations. Energy efficiency standards common in automotive embedded systems were prioritized, along with ISO 26262 compliance to address functional safety in ADAS applications. Ethernet communication conformed to IEEE 802.3 standards for reliable data transfer, and ONNX compatibility was considered for interoperable model deployment. The project adhered to IEEE 730 guidelines for software quality assurance, ensuring high coding and documentation standards.

Revisions and changes were tracked using Git as the version control system, with the project hosted on GitHub. This allowed for efficient version tracking, detailed commit messages, and branching strategy segregated feature development, bug fixes, and final deployment to maintain workflow integrity. Tagged milestones ensured ease of version retrieval and rollback, enhancing development efficiency and reliability.

The project employed robust communication protocols to support testing and deployment. Ethernet (IEEE 802.3) provided high-speed data transfer between the host system and FPGA, while UART facilitated debugging and low-level communication with peripherals. The SD card interface

supported firmware updates and additional storage. Vitis AI runtime leveraged proprietary APIs for seamless communication between the Deep Learning Processing Unit (DPU) and application software. These standards, tools, and protocols ensured the project was not only high-performing but also well-documented and scalable for future enhancements.

V. PROTOTYPE

The project development progressed through multiple iterations of prototypes during the Spring Quarter. Each stage involved testing, refining, and optimizing the lane detection models on FPGA hardware. Below is an outline of the prototypes with labelled figures, captions, and detailed explanations.

A. Prototype 1: Initial Model Implementation (YOLOv3)

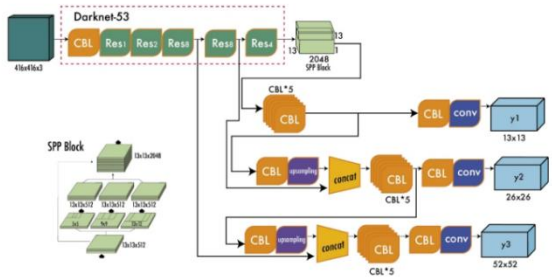

Figure 3: YOLOv3 Lane Detection Model on FPGA

Initial implementation of YOLOv3 lane detection on the Xilinx Kria KV260 FPGA.

The first prototype utilized YOLOv3 for lane detection, known for its general-purpose object detection capabilities. This phase focused on integrating the model with the Vitis AI framework and deploying it on the FPGA. While the model achieved satisfactory lane detection, its relatively low FPS (~10) and accuracy (~90%) indicated a need for optimization.

B. Prototype 2: VPGNet Implementation and Testing

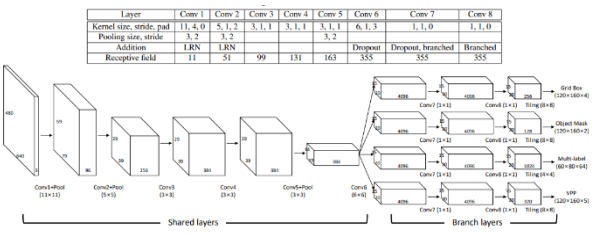

Figure 4: VPGNet Lane Detection under Varied Conditions

Testing VPGNet on the FPGA under different weather conditions.

The second prototype explored VPGNet, a vanishing point-guided network designed for robustness. Tests under various conditions, including rain, snow, and night-time driving, showed improved accuracy (~89%-95%) and higher FPS compared to YOLOv3. VPGNet demonstrated better adaptability to challenging road scenarios, confirming its suitability for real-time applications.

C. Prototype 3: Ultrafast (ResNet18)

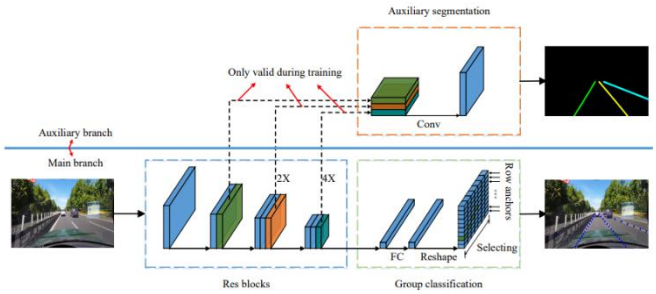

Figure 5: Ultrafast Model Outputs during FPGA Deployment

Deployment of Ultrafast (ResNet18) lane detection on FPGA, showing the highest accuracy.

The third iteration used the Ultrafast model, which achieved the highest accuracy (~95%) across all prototypes. The model's integration involved pruning and quantization using the Vitis AI toolkit, significantly improving execution speed without compromising accuracy. This phase also tested the model's performance under high-resolution (1280×720) video inputs.

D. Prototype 4: Quantized Model Deployment

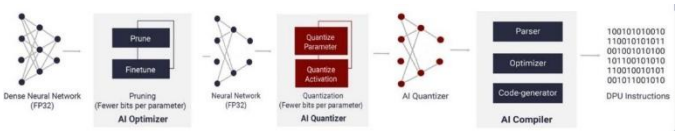

Figure 6: Quantized Model on FPGA with Vitis AI Profiler Results

INT8 quantization results, showing reduced power consumption and enhanced performance.

This prototype involved the quantization of floating-point models to INT8 using the Vitis AI framework. The profiling results indicated a substantial reduction in computational overhead, enabling real-time performance with minimal power consumption. Quantized models delivered faster processing speeds and maintained accuracy for all tested conditions.

E. Prototype 5: End-to-End System Testing

Final Lane Detection System in Real-Time Operation

Caption: The final prototype demonstrating real-time lane detection on FPGA hardware.

Description:

The final prototype integrated the optimized lane detection model into a complete system, tested in real-time scenarios using the TuSimple dataset. The output was validated against ground truth lane markings, achieving the desired balance of accuracy, FPS, and energy efficiency. This version also supported additional testing for robustness under varying environmental conditions.

VI. Experiments

The experiments were conducted using the Xilinx Kria KV260 Vision AI Starter Kit as the hardware platform. The Vitis AI framework was employed for model optimization, quantization, and deployment. Testing utilized the TuSimple dataset, a recognized benchmark for lane detection, with various environmental conditions such as rain, snow, and nighttime driving. Real-time video inputs were provided to the FPGA, and outputs were compared to the ground truth.

A. Software/Hardware & Versions Used

1. Hardware:
   - 1.1. Xilinx Kria KV260 FPGA
   - 1.2. ARM Cortex-A53 processor (4-core)
   - 1.3. 4 GB DDR memory
2. Software:
   - 2.1. Vitis AI 2.5 for model optimization and deployment
   - 2.2. Python 3.9 for pre-processing and testing scripts
   - 2.3. TuSimple dataset for training and validation
   - 2.4. Ubuntu 20.04 as the development host OS

B. Objectives of Test Cases

The primary objectives of the test cases were to:

1. Evaluate the accuracy and robustness of lane detection models (VPGNet, Ultrafast, YOLOv3) across diverse conditions.
2. Measure real-time processing capabilities, including FPS and latency.
3. Validate the energy efficiency of the FPGA implementation against other platforms like NVIDIA Jetson Nano.
4. Ensure compatibility of quantized models with FPGA hardware.

VII. Results

The project successfully showcased the advantages of FPGA-based deployment for real-time lane detection. VPGNet emerged as the most robust model, achieving ~95% accuracy in normal conditions and maintaining strong performance (89%-95%) in adverse scenarios like rain, snow, and nighttime driving. Ultrafast (ResNet18) delivered comparable accuracy (~95%) under daylight conditions but exhibited slightly lower robustness (~90%) in challenging scenarios. YOLOv3, while simpler to implement, achieved ~90% accuracy in normal conditions and dropped to ~85% under adverse conditions, highlighting its limitations.

In terms of real-time performance, VPGNet achieved 32 FPS, and Ultrafast reached 30 FPS, both meeting real-time requirements.

Metrics	VPGNet	Ultrafast	YOLOv3
FPS	43	38	15
Accuracy (%)	89	91	83
Power(W)	7.36	9.56	9.92
Cpu Utilization%	51	81	49
Total Ram Usage%	42	52	57

Figure 7: Model comparison results

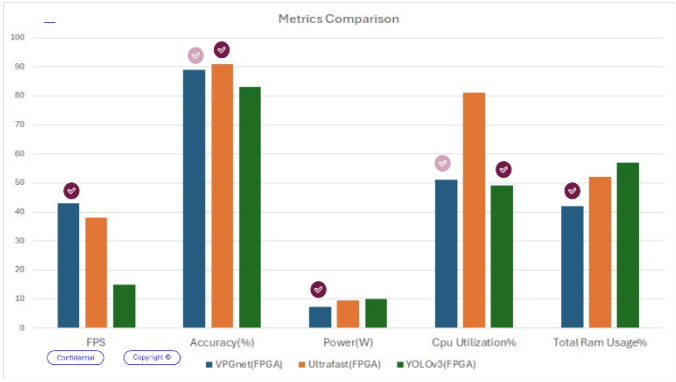

Figure 8: Model comparison chart indicating Ultrafast performance compared to VPGnet and YOLOv3

However, YOLOv3 lagged at 20 FPS, making it less suitable for real-time applications. Energy efficiency testing demonstrated that the Xilinx Kria KV260 FPGA consumed 7.5-15 W during real-time video processing, significantly outperforming the Jetson Nano, which consumed 10-20 W. The quantized versions of VPGNet and Ultrafast showed minimal accuracy loss (<1%) while improving FPS by ~20%, validating quantization as a powerful optimization technique for balancing performance and efficiency.

Metrics	Jetson Nano	Xilinx Kria Kv260
CPU	Quad-core ARM Cortex-A57	Quad-core ARM Cortex-A57
GPU	ARM Cortex-A53 4-core	ARM Cortex-A53 4-core
Memory	NVIDIA Maxwell 128-core GPU	NVIDIA Maxwell 128-core GPU
Storage	Xilinx FPGA (Programmable logic)	Xilinx FPGA (Programmable logic)
Power Supply	4 GB	4 GB

Figure 9: Specification of Nvidia Jetson Nano and Xilinx Kria Kv260

Robustness testing under various weather conditions further highlighted VPGNet's superiority, making it the most reliable choice for real-world applications. The Xilinx KV260 FPGA consistently outperformed the Jetson Nano in energy efficiency and processing speed, underscoring its suitability for embedded vision applications. These results establish FPGA-based systems as a compelling solution for accurate, efficient, and real-time lane detection in safety-critical environments. Future enhancements will focus on improving performance under complex road scenarios like lane merges and extreme weather conditions.

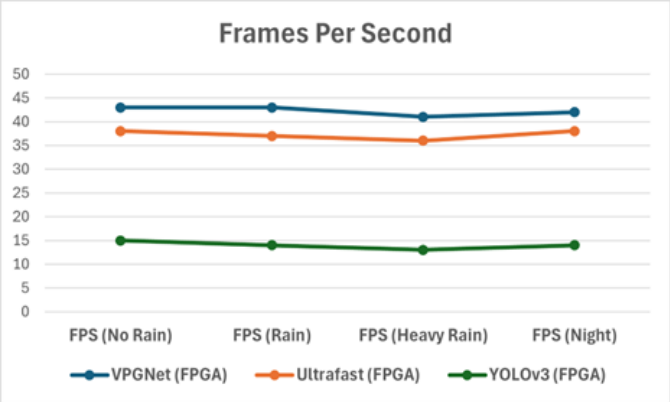

Figure 10: Frame per second chart of the models in different weather conditions.

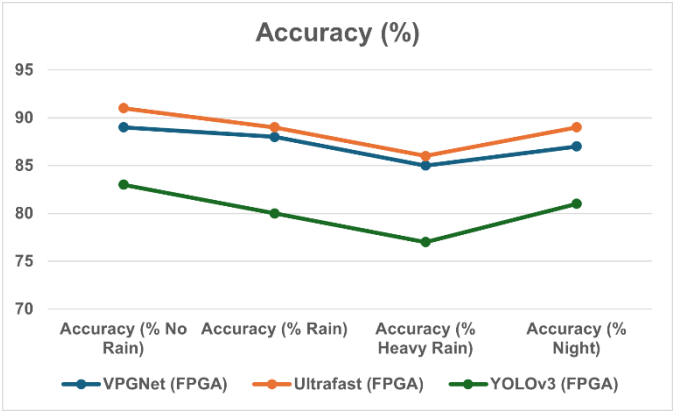

Figure 11: Accuracy chart of the models in different weather conditions.

Figure 12: YOLOv3 Lane detection of on E Peltason Drive, around UC, Irvine Campus.

ACKNOWLEDGMENT

We extend our heartfelt gratitude to Professor Elaheh (Eli) Bozorgzadeh from the Computer Science Department at the University of California, Irvine, for her invaluable guidance and support throughout the project. Her insights and mentorship were instrumental in shaping the direction of this work. We also thank the Department of Embedded and Cyber-Physical Systems for providing the necessary resources and infrastructure to carry out this research. Lastly, we are grateful to our peers and colleagues who offered constructive feedback and encouragement during the various stages of development and testing..

REFERENCES

The template will number citations consecutively within brackets [1]. The sentence punctuation follows the bracket [2]. Refer simply to the reference number, as in [3]—do not use "Ref. [3]" or "reference [3]" except at the beginning of a sentence: "Reference [3] was the first ..."

Number footnotes separately in superscripts. Place the actual footnote at the bottom of the column in which it was cited. Do

not put footnotes in the abstract or reference list. Use letters for table footnotes.

Unless there are six authors or more give all authors' names; do not use "et al.". Papers that have not been published, even if they have been submitted for publication, should be cited as "unpublished" [4]. Papers that have been accepted for publication should be cited as "in press" [5]. Capitalize only the first word in a paper title, except for proper nouns and element symbols.

For papers published in translation journals, please give the English citation first, followed by the original foreign-language citation [6].

[1] G. Eason, B. Noble, and I. N. Sneddon, "On certain integrals of Lipschitz-Hankel type involving products of Bessel functions," Phil. Trans. Roy. Soc. London, vol. A247, pp. 529–551, April 1955. *(references)*

[2] J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

[3] I. S. Jacobs and C. P. Bean, "Fine particles, thin films and exchange anisotropy," in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.

[4] K. Elissa, "Title of paper if known," unpublished.

[5] R. Nicole, "Title of paper with only first word capitalized," J. Name Stand. Abbrev., in press.

[6] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, "Electron spectroscopy studies on magneto-optical media and plastic substrate interface," IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].

[7] M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.

IEEE conference templates contain guidance text for composing and formatting conference papers. Please ensure that all template text is removed from your conference paper prior to submission to the conference. Failure to remove template text from your paper may result in your paper not being published.