SSRG International Journal of VLSI and Signal Processing
ISSN: 2394–2584 / https://doi.org/10.14445/23942584/IJVSP-V10I2P104

Volume 10 Issue 2, 22-26, May-Aug 2023

Original Article

Advanced Driver Assistance System (ADAS) on FPGA

Mayank Kumar[1], A. Niharika[2], Kethireddy Anjali Reddy[3], Harsh Gupta[4], K. N. Pushpalatha[5]

[1,2,3,4,5]*Department of Electronics and Communication Engineering, Dayananda Sagar College of Engineering, Karnataka, India*

Received: 30 April 2023 Revised: 07 June 2023 Accepted: 22 June 2023 Published: 06 July 2023

Abstract *- Advanced Driver-Assistance Systems (ADAS) can help drivers in the driving process and increase driving safety by automatically detecting objects, doing basic classification, implementing safeguards, etc. ADAS integrates multiple subsystems, including object detection, scene segmentation, lane detection, and so on. In this paper, we establish a framework for computer vision features, i.e., lane detection, object detection, object distance estimation and traffic sign recognition of ADAS. Modern machine learning algorithms like Canny edge detection for lane detection and a CNN-based approach are used for object detection. The system deployed aims to achieve higher (Frames Per Second) FPS for one channel of 55 FPS. The performance of FPGA is optimized by software and hardware co-design. Realization on the DE-10 Nano board with Cyclone V FPGA and a dual-core ARM Cortex A9, which meets real-time processing requirements. An increasing amount of automotive electronic hardware and software involves significant changes in the modern automobile design process to address the convergence of conflicting goals - increased reliability, reduced costs, and shorter development cycles. The prospectus to tackle car accident occurrences is making ADAS even more critical. This paper proposes an efficient solution for ADAS on FPGA.*

Keywords *- ADAS, CNN, FPGA, FPS, Machine Learning.*

1. Introduction

Safety improvements for automobiles in the past such as shatter-resistant glass, three-point seatbelts, and airbags - were safety measures adapted to mitigate accidents. Now, ADAS systems actively improve safety with the help of modern technology like embedded vision. [1]

Automobiles are the foundation of the next generation of mobile-connected devices, with rapid advancements in autonomous vehicles. Autonomous applications are divided into different chips, called systems on a chip (SoCs). These chips connect sensors to actuators through interfaces and high-performance electronic controller units (ECUs).

Self-driving cars use various applications and technologies to gain 360-degree vision, both near and far. That means hardware designs are using more advanced process nodes to meet ever-higher performance targets while reducing power and footprint demands. Conventionally, ADAS had been implemented on application-specific complex impromptu solutions built especially for automotives since they suppose a large market.[2]. This research aims to deploy an FPGA for implementing ADAS.

Owing to the restrictions of software for allowing a real-time embedded system to be implemented An FPGA board is required for a level 5 autonomous vehicle, particularly one employing complex vision tasks.[3]

This paper proposes an efficient solution for ADAS on FPGA. Specifically, the objectives of this work are listed as follows. (1) We propose a framework for ADAS which integrates the tasks of lane detection, object detection, distance estimation, and traffic light recognition and achieves comparable precision with task-specific models. (2) We optimize the system performance through the joint optimization of software and hardware. The system will be deployed on Altera Cyclone V FPGA to achieve 55 FPS for one channel.

2. Reference Study

All existing car automation techniques use microprocessors. The microprocessors used are typically embedded systems designed for automotive applications. These microprocessors had to meet the stringent requirements of real-time processing, reliability, and safety criticality. With the advent of FPGAs in the 1980s, they gained popularity because of their reprogramming ability and parallel processing capabilities.

Lane detection is done using edge detection algorithms. Canny Edge Detection and Sobel Edge Detection are the most used edge detection algorithms.

The former is often considered superior to the latter due to several reasons like Accurate Edge Localization and Automatic Thresholding. This paper uses the Canny edge detection algorithm.

Traffic light recognition is part of an autonomous system that is supposed to take appropriate decisions on-road—traffic light detection in complex urban environments. A combination of colour-based segmentation, pattern matching, and machine learning techniques like the YOLOv5 algorithm was employed to detect traffic lights.

3. Materials and Methods

One way to implement ADAS is to use an FPGA as the primary hardware platform. An FPGA is a type of programmable hardware that can be configured to perform a wide range of digital functions. It is often used in applications where flexibility and fast response times are important.

The processing unit in an ADAS system is responsible for analysing the information from the sensors and making decisions based on that. Sometimes, this may involve running complex algorithms or machine learning models to interpret the data and determine the best course of action. The processing unit may be implemented using a microcontroller, a processor, or an FPGA, depending on the application's specific requirements. The FPGA used in this project is the Intel DE10 Nano. The DE10-Nano is a compact, low-power development board featuring an Intel Cyclone V FPGA. The DE10-Nano is equipped with a dual-core Arm Cortex-A9 processor, which can be used to run software applications.

Lane detection is a technology used in self-driving cars ADAS to identify the location and boundaries of lanes on the road. This paper uses the Canny edge detection [4] algorithm for detecting lanes, as described in Figure 2.

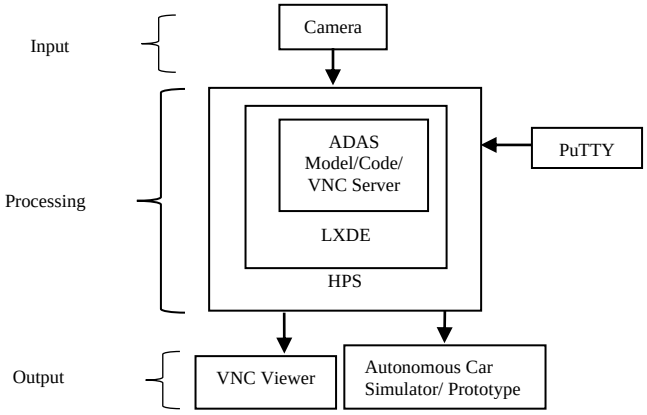

Fig. 1 Block Diagram

The Hough transform [5] is an image-processing technique used in edge detection to improve the accuracy and robustness of lane detection. It is used in a variety of other applications, such as detecting circles, ellipses, and other geometric shapes in images.

The breakthrough of neural networks is that object detection no longer must be a hand-crafted coding exercise. Neural networks allow features to be learned automatically from training examples.[6]

The pipeline for object detection and distance estimation is seen in Figure 3. This research uses the YOLOv5[7] architecture for object detection. Object detection, a use case for which YOLOv5 is designed, involves creating features from input images. These are passed through a prediction system to draw boxes called bounding boxes around objects and predict their classes.

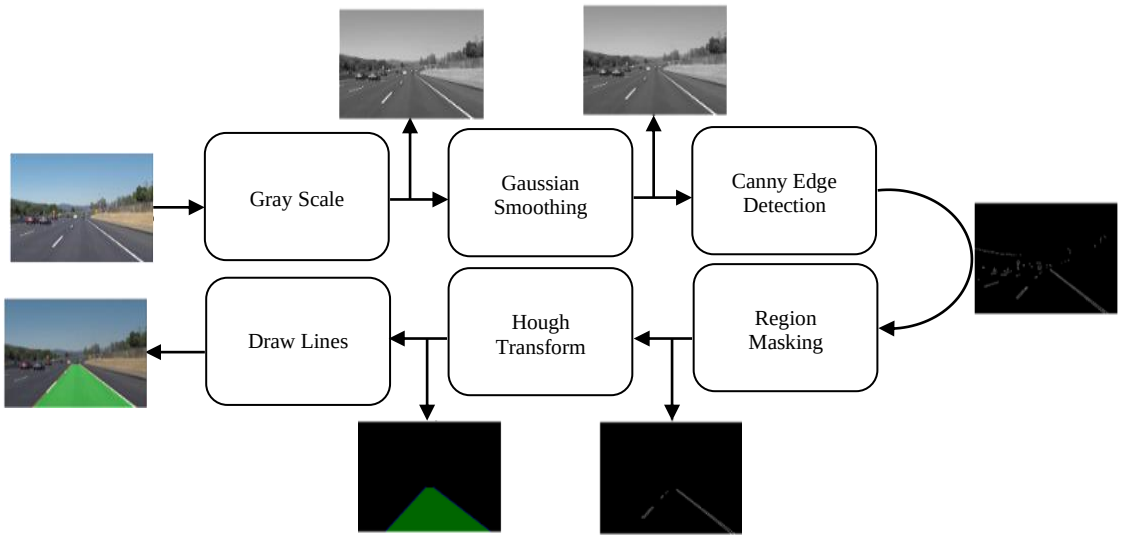

Fig. 2 Lane detection pipeline

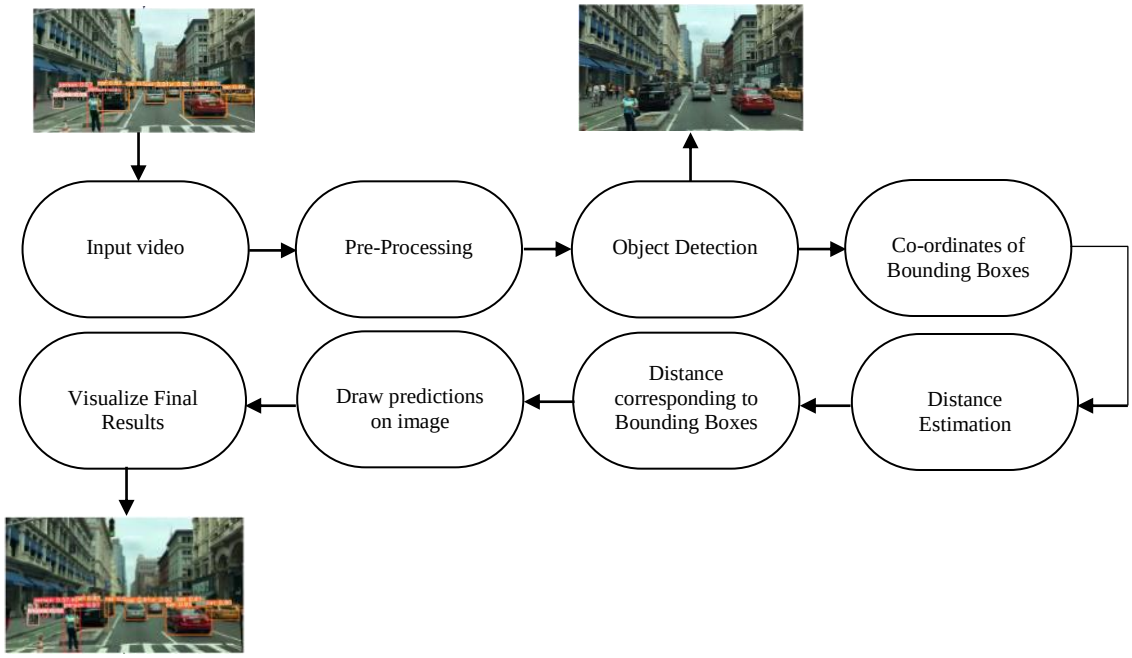

Fig. 3 Object detection pipeline

An older version of YOLO introduced the idea of learning anchor boxes based on the distribution of these boxes in the custom dataset with K-means and genetic learning algorithms. This is very important for custom tasks because the set of bounding box sizes and locations may be dramatically different from the pre-set bounding box anchors in another dataset. The YOLOv5 network predicts bounding boxes as deviations from a given set of anchor box dimensions to make box predictions. This model is trained on the FLIR dataset.

The distance between the camera and a target object in the surrounding environment is determined using size-based distance approximation.[9] This model is trained on the KITTI dataset.

4. Results and Discussion

4.1. Lane Detection

Figure 4 is the snapshot of an input fed to the lane detection pipeline. Figure 5 is the final output. The lane detection feature is used to detect the lane where the car is currently present and the ones that may exist on either side of the lane. The green overlay on the input gives a clear understanding of the path ahead of a driver. This feature has increased importance when it is integrated into an autonomous driving system [10] wherein the car is itself expected to take decisions. The lane departure warning system provides a physical alarm to the driver about the departure from the existing lane.

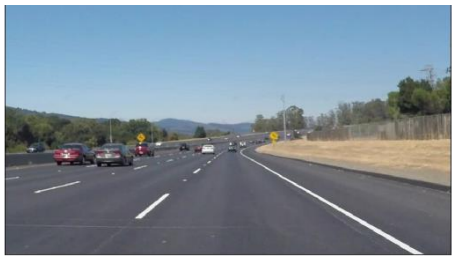

Fig. 4 Input to line detection pipeline

Fig. 5 Lane detected output

CARLA[11][13] was used to visualize these results in a real-time environment.

Fig 6 Visualization of CARLA

4.2. Object Detection, Object Distance Estimation, Traffic Light Detection

Figure 7 describes the features of object detection, object distance estimation and traffic light detection. The bounding boxes have different colours for different objects – orange for cars, red for people, etc. At the header of these bounding boxes are the identified objects' labels and a number that lies between 0 to 1 called objectness. Objectness indicates the accuracy with which the algorithm has identified the object.

Further, a green bounding box enclosing a traffic signal signaling 'green for go' is also seen. In the case of traffic light detection, the bounding box labels the signal identified.

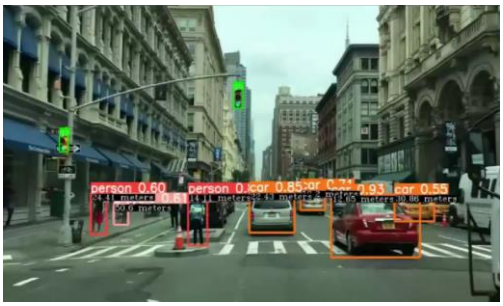

Fig. 7 Object detection, distance estimation, traffic light detection

5. Conclusion

The implementation of ADAS on FPGA comes with advantages like parallel processing capabilities of FPGA[14], low power consumption and better throughput, along with a great deal of cost reduction.

The improvement in the performance of the lane detection system due to pre-processing on FPGA can be seen in Table 1.

Table 1. Comparison Results

Factor	FPGA	Processor
Frames per Second	81.3	27.01
Time for one image frame	36.99ms	77.84ms

The programming language for pre-processing is Verilog hardware description language. The function works in 12.3 ms. This means that the performance is 81.3 fps**.** This result is 3.01 times better than the computer solution (2.5 GHz Intel Core i5) and 18.4 times better than the ARM Cortex A9 solution (667 MHz).

According to the August 2016 Traffic Safety Facts Research Note by the National Highway Traffic Safety Administration (NHTSA)[15], "The Nation lost 35,092 people in crashes on U.S. roadways during 2015." This 7.2% increase was "the largest percentage increase in nearly 50 years." An analysis revealed that about 94% of those accidents were caused by human error. The application of this project is a stepping stone towards bringing a transformation in the automobile industry towards safer roads and safer pedestrians. The work carried out in this paper has brought to light, promising areas of further research [16] in the ADAS system optimization field. Further improvements by deep model compression and software-hardware collaborative optimization[16] can be obtained.

References

[1] The Synopsys website, 2023. [Online]. Available: https://www.synopsys.com/designware-ip/technical-bulletin/deep-learning-dwtb-q217.html

[2] Juan Borrego-Carazo et al., "Resource-Constrained Machine Learning for ADAS: A Systematic Review," *IEEE Access*, vol. 8, pp. 40573-40598, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[3] Hamish Simmonds et al., "Autonomous Vehicle Development Using FPGA for Image Processing," *2019 International Conference on Field-Programmable Technology*, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[4] Ghassan Mahmoud Husien Amer, and Ahmed Mohamed Abushaala, "Edge Detection Methods," *2015 2nd World Symposium on Web Applications and Networking*, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[5] A Complete Guide On Hough Transform, 2022. [Online]. Available: https://www.analyticsvidhya.com/blog/2022/06/a-complete-guide-on-hough-transform/

[6] YOLO: Algorithm for Object Detection Explained [+Examples], 2023. [Online]. Available: https://www.v7labs.com/blog/yolo-object-detection

[7] M. Karthi et al., "Evolution of YOLO-V5 Algorithm for Object Detection: Automated Detection of Library Books and Performace validation of Dataset," *2021 International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems*, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[8] M.R. Ezilarasan, and J. Brittopari, "An Efficient FPGA-Based Adaptive Filter for ICA Implementation in Adaptive Noise Cancellation," *SSRG International Journal of Electrical and Electronics Engineering*, vol. 10, no. 1, pp. 117-127, 2023. [CrossRef] [Publisher Link]

[9] Shi-Huang Chen, and Ruie-Shen Chen, "Vision-Based Distance Estimation for Multiple Vehicles Using Single Optical Camera," 2011 Second International Conference on Innovations in Bio-inspired Computing and Applications, 2011. [CrossRef] [Google Scholar] [Publisher Link]

[10] Abdulhakam AM. Assidiq et al., "Real time Lane Detection for Autonomous Vehicles," *2008 International Conference on Computer and Communication Engineering*, 2008. [CrossRef] [Google Scholar] [Publisher Link]

[11] CARLA Simulator. [Online]. Available: https://carla.readthedocs.io/en/latest/download/

[12] E. Ganesan, and V. Sakthivel, "A Novel FPGA Design with Hybrid LUT / MUX Architecture," *SSRG International Journal of Electronics and Communication Engineering*, vol. 3, no. 11, pp. 6-8, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[13] Hyeonjae Jeon et al., "CARLA Simulator-Based Evaluation Framework Development of Lane Detection Accuracy Performance Under Sensor Blockage Caused by Heavy Rain for Autonomous Vehicle," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 9977-9984, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[14] Alexander Stumpf et al., "Benefits of an FPGA based SRM Controller," *2014 9th IEEE Conference on Industrial Electronics and Applications*, 2014. [CrossRef] [Google Scholar] [Publisher Link]

[15] Traffic Safety Facts. [Online]. Available: https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/812318

[16] Issam W. Damaj, Jibran K. Yousafzai, and Hussein T. Mouftah, "Future Trends in Connected and Autonomous Vehicles: Enabling Communications and Processing Technologies," *IEEE Access*, vol. 10, pp. 42334-42345, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[17] Ahmad Shawahna, Sadiq M. Sait, and Aiman El-Maleh, "FPGA-Based Accelerators of Deep Learning Networks for Learning and Classification: A Review," *IEEE Access*, vol. 7, pp. 7823-7859, 2019. [CrossRef] [Google Scholar] [Publisher Link]