Design Flow of mmWave Radar and Machine Vision Fusion for Pedestrian Collision Warning

Kaitlyn Graves, Madiha Kanwal, Xinrui Yu, and Jafar Saniie
Embedded Computing and Signal Processing (ECASP) Research Laboratory (http://ecasp.ece.iit.edu)
Department of Electrical and Computer Engineering
Illinois Institute of Technology, Chicago IL, U.S.A.

***Abstract*—This paper presents a millimeter wave (mmWave) radar and Machine Vision fusion system to alert drivers of potential pedestrian collisions. The system is composed of two subsystems, the mmWave Pedestrian Localization subsystem and the Machine Vision Pedestrian Classification subsystem. The mmWave Pedestrian Localization subsystem obtains the relative location of the pedestrians using a mmWave radar sensor while the Machine Vision Pedestrian Classification subsystem uses Histogram of Oriented Gradients and Support Vector Machine algorithms to classify pedestrians in a camera's field-of-view. The two-layer pedestrian detection design protects the system from any miss detections within a single subsystem. By utilizing mmWave technology with Machine Vision, the safety operation of cars and pedestrian safety can be increased. The proposed system utilizes Texas Instrument's AWR1642BOOST mmWave Radar with the high computing power of NVIDIA's Jetson Nano.**

***Keywords*—mmWave radar, machine vision, pedestrian collision warning**

I. Introduction

From 2010 to 2019, there has been a 46% increase of pedestrian fatalities in traffic-related accidents [1]. Advanced Driver Assistance Systems (ADAS) are being integrated into many cars now to help prevent accidents like those. The goal of these systems is to help decrease human error and increase driving safety operations [2]. Sensors are used in ADAS to detect possible obstacles around a car and report them to the driver. An example of a sensor used for obstacle detection is a camera. Nearby obstacles can be identified by a camera; however, a lot of processing goes into determining relative distances of objects from a camera image which complicates the need for real-time results of ADAS systems [3]. Also, the field of view of cameras can be obstructed by environmental effects, such as fog or lighting [4].

Other technologies, such as radars, are better suited for determining relative distances of obstacles and are less susceptible to visual obstacles such as rainy weather [5]. Millimeter wave (mmWave) radars are commonly used for autonomous vehicle applications. They can detect objects and obtain their position and velocity by mapping the reflections of transmitted, high-frequency electromagnetic waves. Due to their high frequency operation, mmWave radars have high precision all the way down to a fraction of a millimeter [6][7]. While mmWave radars can detect obstacles' locations accurately, it can be difficult to classify the type of object that is being measured. By pairing a mmWave radar with a camera, a system can classify objects as well as obtain their relative position easily and accurately.

This paper proposes a system that uses a mmWave Radar with Machine Vision for Pedestrian Collision Warnings. The system uses image processing and machine learning to classify pedestrians, and it uses mmWave object point clouds to map their accurate relative position. The mmWave Pedestrian Localization subsystem and Machine Vision Pedestrian Classification Subsystem are fused into a two-layer detection design to protect against miss detections.

II. System Design

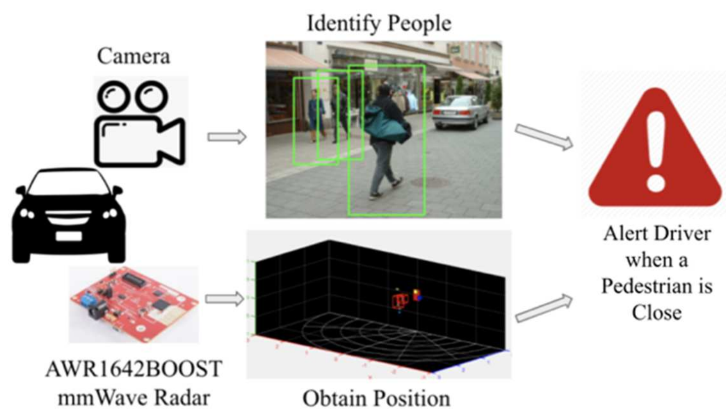


Fig. 1. Pedestrian Collision Warning System Overview

A. Overall

The Pedestrian Collision Warning (PCW) Fusion System combines mmWave radar technology with image processing and machine learning to alert drivers of nearby pedestrians. The PCW system is made up of two subsystems: the Machine Vision Pedestrian Classification (MVPC) subsystem and mmWave Pedestrian Localization (mmWPL) subsystem. The MVPC subsystem uses a pre-trained Histogram of Oriented Gradients and Support Vector Machine model to classify people in an image. The mmWPL subsystem uses a mmWave radar to obtain accurate locations for a detected target. The PCW Fusion System makes a decision based on the two-layer design of the subsystems on whether to produce an alert to the driver. Fig. 1 displays an overview of the PCW Fusion System.

B. Hardware Components

Texas Instruments' AWR1642BOOST-ODS Evaluation Module (EVM) was selected as the mmWave radar for the system. The on-board AWR1642 radar-on-chip is a Frequency Modulated Continuous Wave Radar. The EVM operates at 77-81 GHz which is the frequency range reserved for vehicular use. It has a C674x Digital Signal Processor and ARM Cortex-R4 Processor, so all signal processing can be performed on-board. [8]

The antenna array structure of the AWR1642BOOST-ODS model compared to the regular AWR1642BOOST is set up to capture a wider field-of-view which is beneficial for detecting pedestrians at a wider angle [8]. Using a UART connection, a computer can send configuration information to the EVM and the EVM can send a data stream of sensor collected information to the computer.

The PCW Fusion System uses the NVIDIA Jetson Nano, a single board computer optimized for image processing and artificial intelligence. The Jetson Nano is chosen due to its efficient parallel processing capabilities on account of its 128-core Maxwell GPU, making it ideal for high computational tasks [9]. The mmWave Radar and a standard web camera is connected to the Jetson Nano through USB 3.0. The Jetson Nano runs the PCW Python Program and displays the Collision Warning through the terminal.

C. mmWPL Subsystem

Texas Instrument's AWR1642BOOST-ODS is used to obtain the locations of the pedestrians in the field-of-view. The AWR1642BOOST-ODS is running TI's "16xx - People Counting" Demonstration Program to locate and track pedestrians [10]. The program performs digital signal processing on-board by implementing detection and tracking algorithms. Fig. 2 summarizes the demonstration program as a flowchart. A Python program is used to collect the EVM's output data stream.

1) AWR1642BOOST Pedestrian Tracking: The low-level signal processing is performed in the Evaluation Board's C674x DSP. During this step, point cloud data is collected with parameters such as range, azimuth, Doppler estimation, and Signal-to-Noise Ratio (SNR) information. Range processing is completed by running a 1D Windowing and 1D FFT on the digital RF input from each Rx antenna. Static clutter removal is performed to remove any DC, or static, components. Capon Beamforming, an angle of arrival algorithm, is used to obtain the azimuth element for the data. A generated range and azimuth heatmap is used for Two Pass CFAR point object detection. Two pass Constant False Alarm Rate (CFAR) detects the points by first performing cell-averaging smallest of CFAR in the range domain then in the angle domain. [11]

The grouping and tracking of the point cloud data is executed in the ARM Cortex-R4 Processor. Individual targets are detected by a process called Allocation. Allocation is performed by clustering detected points based on a minimum SNR threshold and minimum number of points in proximity to each other. If a group of points meets the requirements, it becomes a target. Targets are tracked by predicting its centroid location using state and process covariance matrices based on time n-1. The output from the high-level processing is a target list with information of its Cartesian position and Cartesian velocity. [11]

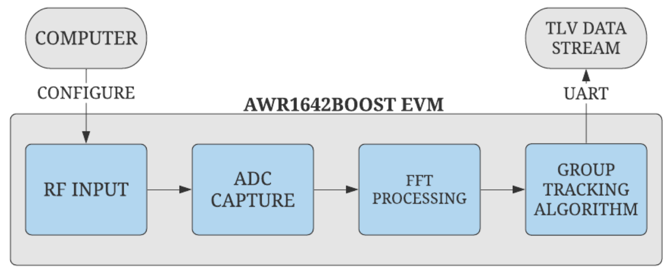


Fig. 2. AWR1642BOOST Pedestrian Tracking Flowchart [11]

2) Python Data Extraction: The EVM outputs the data in TLV (Type-Length-Value) packets which has a structure shown in Fig. 3. The frame header is a constant 52 bytes and contains information such as packet length, timestamp, and track processing time. There are three possible TLV packets: Point Cloud, Target, and Target Index. The type is specified in the packet header. Point Cloud TLVs contain the information of the detected point cloud objects such as range, azimuth, and SNR. Target TLVs contain the information of the tracked targets, such as Target ID and Cartesian position and velocity. The Target Index TVL reports the Target IDs.

Fig. 3. Frame Format of AWR1642 Output Datastream [11]

For the proposed PCW system, the Cartesian position of the Target TLV is the parameter of focus. A Python program based on [12] was created to extract the necessary data from the EVM's frames as shown in Fig. 4. The program communicates with the EVM by establishing serial Data and Configuration ports. If present, the Target TLV data is read from inside the buffer and stored into a dictionary for ease of access. The ranges from the EVM to the Targets are calculated using the Pythagorean Theorem with the obtained Cartesian position.

$$range = \sqrt{x^2 + y^2} \tag{1}$$

The mmWPL Subsystem returns a list format:

$$[targetID,\ range,\ numTargets] \tag{2}$$

where *targetID* is the unique ID of identified targets, *range* is the calculated range from the EVM, and *numTargets* is the total number of targets detected in that frame.

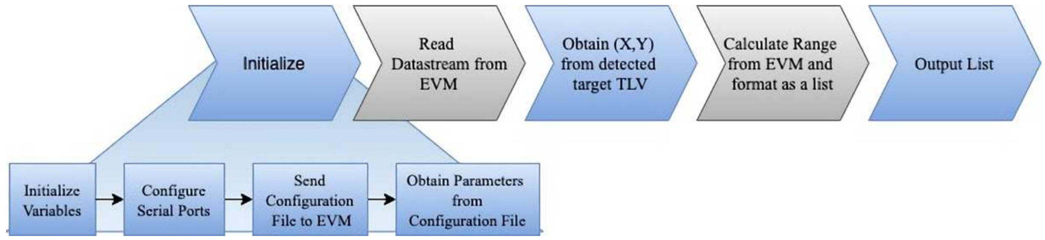
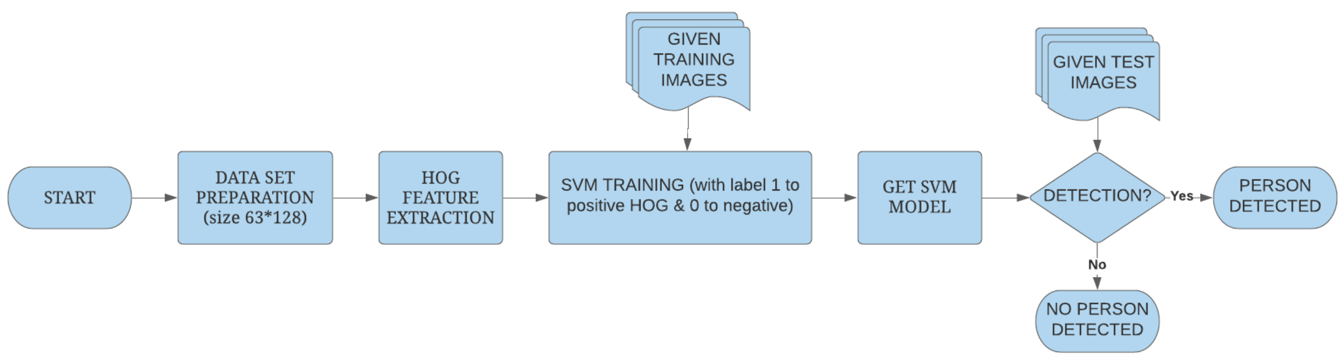


Fig. 4. mmWave Python Script for Data Extraction

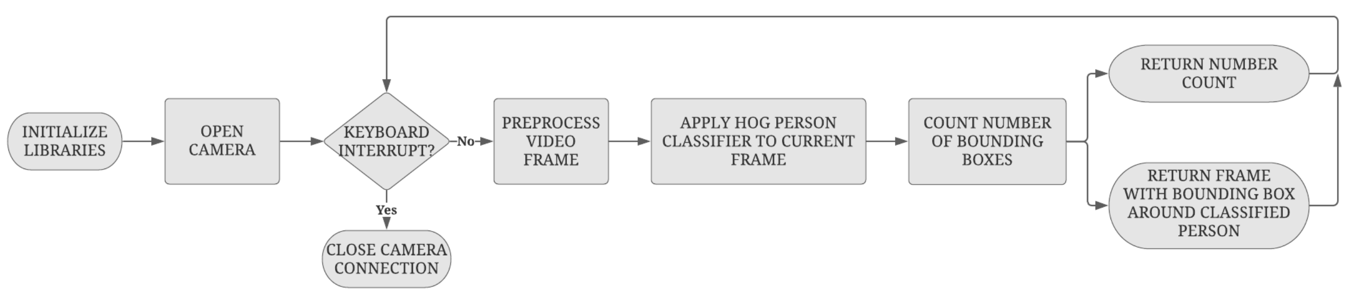
D. Machine Vision Pedestrian Classification Subsystem

The Machine Vision Pedestrian Classification (MVPC) subsystem can classify people using a Histogram of Oriented Gradients (HOG) algorithm and a Linear Support Vector Machine (SVM) model. The Histogram of Oriented Gradients algorithm extracts features using a sliding window to calculate the gradients of the image section, or cell, and create a histogram with the value being gradient angles and weight being gradient magnitude. All the cells' histograms are combined to create a single histogram. The single histogram is fed into the trained Linear SVM model which will decide if a person is detected or not [13]. Fig. 5a displays the training process for the SVM model. The MVPC subsystem uses OpenCV's pre-trained SVM model with the HOG people detector in order to count the number of people in the camera frame as shown in Fig. 5b.

There may be a privacy concern with the MVPC subsystem since the camera is capturing images of people; however, this concern does not apply here since the operational system is not saving the camera feed. The operational system will only use the camera feed to classify if a person is present, then dumps the frame. The data will not be saved anywhere.



(a)



(b)

Fig. 5. (a) SVM HOG Model Training and (b) Machine Vision Pedestrian Classification Flowchart

E. Fusion of Subsystems

1) Warning Decisions: The warning decision ranges were based on a car traveling at 25 mph because that is the speed usually enforced in areas with high pedestrian activity. The four different zones were calculated from statistics given by The National Association of City Transportation Officials and University of Pennsylvania [14]. The 'Perception Reaction Distance' zone is the distance a car travels from when a scenario occurs and the time the driver goes to react, about 16.7 meters. The 'Stopping Distance' is the distance it takes the vehicle to brake, about 8.3 meters. The 'Warning Zone' was chosen as a 5 meter space after the 'Perception Reaction' and 'Stopping

Distance' zones. Any distance after the 'Warning' zone is considered 'Safe'. Due to testing environment constraints, the zones are scaled by 0.1 to obtain experimental results. Each zone produces a different warning output. The decision zones are summarized in Fig. 6.

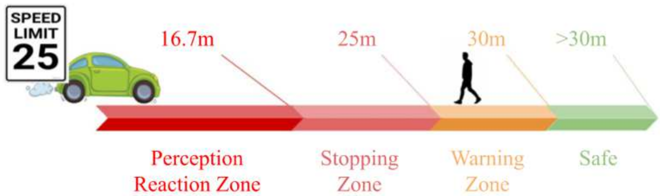


Fig. 6. Warning Decision Zones

2) Two-Layer Detection: The mmWave Localization and Machine Vision Classification subsystems are combined to create the Pedestrian Collision Warning System. The PCW is a two-layer protected detection system. The mmWave Localization is the first layer. It uses the ranges of the detected targets to determine what warning to output. The second layer is the Machine Vision subsystem which acts as a fail-safe to the mmWave radar. If no targets are detected by the mmWave radar, the total number of pedestrian classifications from the MV subsystem is checked. If the number is greater than zero, a caution warning is issued. Otherwise, the status is safe. The flowchart in Fig. 7 outlines the two-layer system. The warning outputs are summarized in Table I.

TABLE I. SUMMARY OF WARNING OUTPUTS FOR SCALED DISTANCE MEASUREMENTS

Zone	Warning Output
Perception Reaction (<1.67m)	Warning: Pedestrian Extremely Close!
Stopping (1.67<range<2.5m)	Warning: Pedestrian Close By
Warning (2.5<range<3m)	Caution: Pedestrian Close By
Safe (>3m)	Safe
Person classified and no target detected by mmWave	Caution: Pedestrian Detected on Camera
No person classified and no target detected by mmWave	Safe

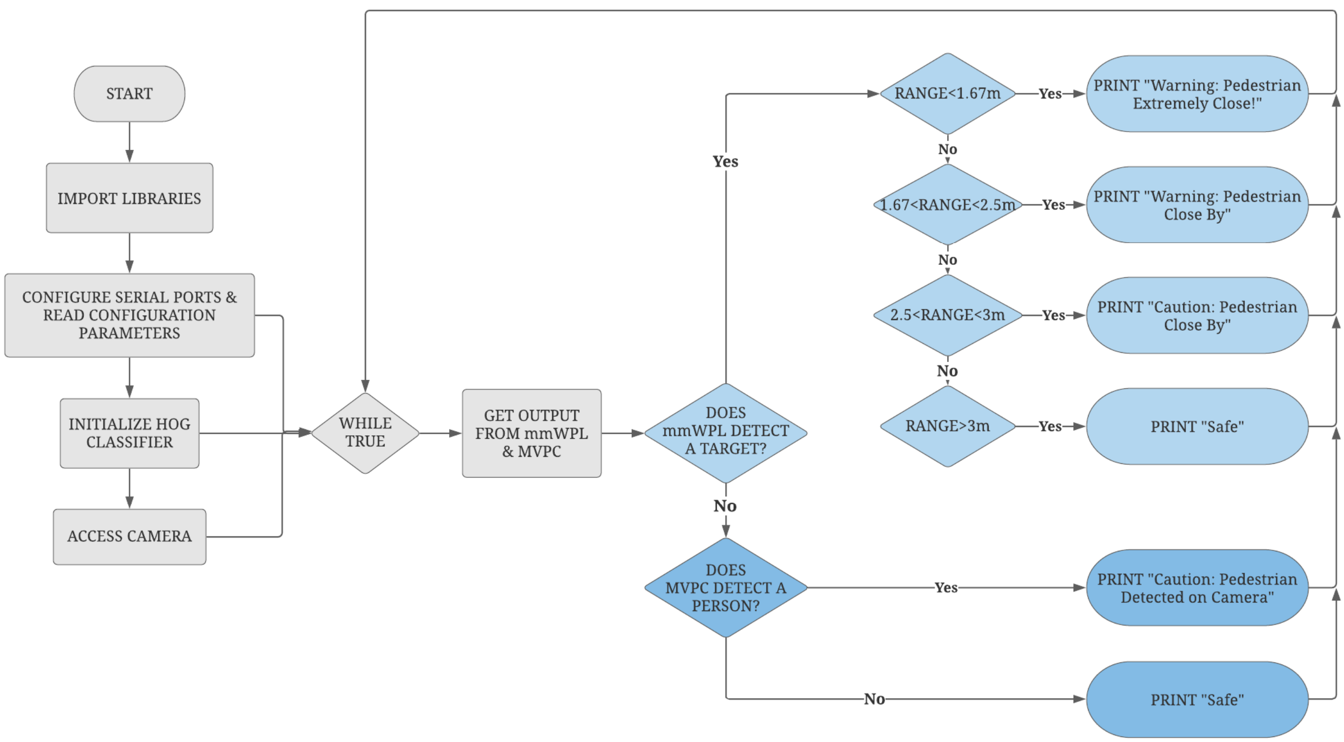


Fig. 7. Pedestrian Collision Warning Fusion System Flowchart

III. Experimental Results

A. mmWave Pedestrian Localization Subsystem

To verify the functionality of the mmWave Pedestrian Localization Subsystem, the EVM was connected to the Jetson Nano that was running the Python Data Extraction program. A subject walked around the room randomly. The returned list of Target IDs, ranges, and number of Targets was printed to the terminal for visualization purposes. The terminal output shows the correct number of Targets being detected and accurate ranges. However, it was observed that there were times when miss detections would occur and no targets are detected even though there was a target in the field of view.

B. Machine Vision Pedestrian Classification Subsystem

To verify the accuracy of the Machine Vision Pedestrian Classification subsystem, a subject moved randomly around a room. It was observed that the system worked best when the full body of the subject was in the camera's field-of-view. If the subject was too close, multiple body parts would be detected instead of the body as a whole. This will not affect the output of the PCW system since the MVPC subsystem only checks for more than zero detections.

C. Pedestrian Collision Warning Fusion System

The PCW system was tested by having a subject stand in each of the marked zones and observing the terminal response. The proper warnings were displayed for each zone as seen in Fig. 8. However, the mmWave radar had more miss detections in the PCW system than it did in the individual subsystem. The MVPC subsystem was able to successfully cover the mmWave radar's missed detections, just as the PCW Fusion system was designed to do.

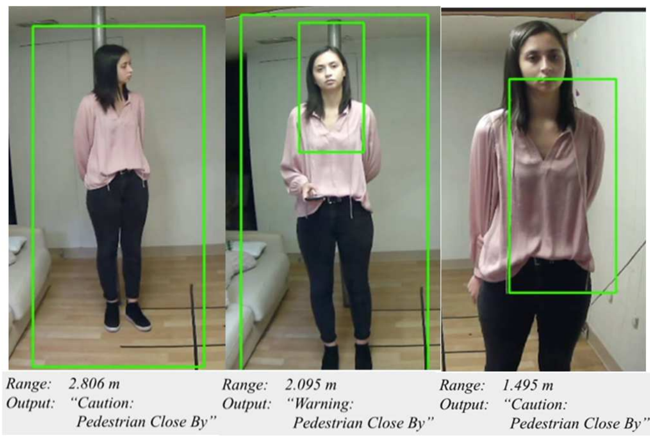


Fig. 8. PCW System Results for the Scaled Warning Zone (left), Stopping Zone (center), and Perception Reaction Time Zone (right)

IV. DISCUSSION

The full capability of the PCW system was not able to be tested due to testing environment and resource constraints. The system had to be tested indoors due to the current dependency on wall outlet power; however, the indoor testing space was limited in size. Also, testing personnel was limited to one person due to COVID-19 restrictions so scenarios such as multiple people in the radar and camera field-of-view could not be tested.

After performing the tests, it was observed that there is a real-time constraint on the MVPC subsystem which slows down the PCW system. It takes approximately 0.6 seconds to process a video frame with the HOG classifier. In 0.6 seconds, a car going 25 mph would travel about 6.7 meters. This is 80% of the total stopping distance. In an application dealing with preventing pedestrian collisions with vehicles, real-time is critical.

As mentioned above, the mmWave subsystem in the PCW had more miss detections than when run individually. A possible cause is that the real-time delay of the MVPC subsystem is affecting how the AWR1642's output is being read. A solution would be to run the two subsystems in parallel and thread their output together to make a warning decision. A quick test performed after the official demonstration proved the mmWave Localization subsystem had fewer miss detections running in parallel than in series in a fused Python script.

V. CONCLUSION

In this paper, a system to utilize mmWave radar technology with Machine Vision for Pedestrian Collision Warning was created. The mmWave Radar Pedestrian Localization Subsystem provides accurate relative position of pedestrians while the Machine Vision Pedestrian Classification Subsystem uses Histogram of Oriented Gradients and a trained Linear SVM model to classify the pedestrians. The subsystems are fused together to create a two-layer detection and warning system.

While the testing results determine the mmWave Radar is producing a large amount of miss detections in the current design, the PCW system shows the benefits of using both mmWave and Machine Vision. The two-layer pedestrian detection design allows for the Machine Vision subsystem to operate as a fail-safe to the mmWave subsystem. Therefore, less miss detections occur, and warnings can be given accurately.

In the future, the increased rates of miss detections will be addressed by redesigning how the PCW system fuses the mmWave Localization subsystem and MVPC subsystem. Parallel architecture will be researched as it was observed to decrease the amount of miss detections the mmWave subsystem creates. In addition, the warning output will be switched from a printed statement in the terminal to a physical output such as an LED, alarm, or automatic braking so the driver of the vehicle can better focus on driving instead of checking the screen. The physical output will also allow the system to be headless for implementation on a vehicle. Another area of interest to explore is comparing different machine learning and deep learning algorithms for the MVPC subsystem. Parameters such as runtime and accuracy can be observed and compared.

REFERENCES

[1] A. Snider. "Pedestrian Deaths Soar in 2020 DESPITE Precipitous Drop in Driving During Pandemic." *GHSA*, Governors Highway Safety Association, 20 May 2021.

[2] A. Paul, R. Chauhan, R. Srivastava, and M. Baruah, "Advanced Driver Assistance Systems," SAE Technical Paper 2016-28-0223, 2016, https://doi.org/10.4271/2016-28-0223.

[3] D. Gerónimo, A. M. López, A. D. Sappa and T. Graf, "Survey of Pedestrian Detection for Advanced Driver Assistance Systems," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32, no. 7, pp. 1239-1258, July 2010, doi: 10.1109/TPAMI.2009.122.

[4] Y. Chen, W. Li, C. Sakaridis, D. Dai and L. Van Gool, "Domain Adaptive Faster R-CNN for Object Detection in the Wild," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3339-3348, doi: 10.1109/CVPR.2018.00352.

[5] V. K. Kukkala, J. Tunnell, S. Pasricha and T. Bradley, "Advanced Driver-Assistance Systems: A Path Toward Autonomous Vehicles," in IEEE Consumer Electronics Magazine, vol. 7, no. 5, pp. 18-25, Sept. 2018, doi: 10.1109/MCE.2018.2828440.

[6] C. Iovescu, S. Rao, "The fundamentals of millimeter wave radar sensors." Texas Instruments. [online]

[7] S. Sun, A. P. Petropulu and H. V. Poor, "MIMO Radar for Advanced Driver-Assistance Systems and Autonomous Driving: Advantages and

Challenges," in *IEEE Signal Processing Magazine*, vol. 37, no. 4, pp. 98-117, July 2020, doi: 10.1109/MSP.2020.2978507.

[8] Texas Instruments, “AWR1642BOOST Product.” [online]

[9] NVIDIA, “NVIDIA Jetson Nano Developer Kit.” NVIDIA Developer. [online]

[10] Texas Instruments, “16xx - People Counting” mmWave Sensors Industrial Toolbox - 4.1.0, TI. [online]

[11] A. Killedar, W. He, G. Akash, “mmWave Radar - IWR1642 People Counting Demonstration.” TI, 2017. [online]

[12] ibaiGorordo,“AWR-1642-Read-Data-Python-MMWAVE-SDK-2.” Github, 2020. [online]

[13] TheDataFrog, “Real-Time Human Detection with OpenCV.” *The Data Frog*, 2019. [online]

[14] University of Pennsylvania School of Engineering “Vehicle Stopping Distance and Time.” National Association of City Transportation Officials, 2013.