ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

UTEM NAVIGATION SYSTEM: PEDESTRIAN AND TRAFFIC SIGN DETECTION USING CNN ALGORITHM

WAN MOHD YAAKOB WAN BEJURI[1] , MUHAMMAD HARRAZ HARUN[2] AND ABDUL KARIM MOHAMAD[1]

[1]Faculty of Information and Communication Technology, Universiti Teknikal Malaysia Melaka, MALAYSIA

[2]Institute of IR 4.0, Universiti Kebangsaan Malaysia, MALAYSIA

E-mail: yaakob@utem.edu.my

ABSTRACT

Navigation is a common problem for all drivers, especially university visitors. Unfamiliar place making the driver become careless and unaware, which give hazard to pedestrians and driver itself. Thus, this system aims to solve the problems, by developing mobile navigation with safety features by taking UTeM campus as our scope of the study. The system using algorithm using CNN as an algorithm and the architecture used is Tiny-YOLOv2 to detect traffic signs and pedestrians. To begin, the dataset containing Person and Traffic Sign images and their annotations will first need to be acquired. Then, the CNN model will be trained and tested. As a result, our proposed system shows that the mean average precision for both classes can achieve as 90.44%, when it is implemented in a conventional smartphone. This is proof that our system can provide better capability when it is implemented with a smartphone device. Thus, it contributes to being a new mobile navigation system that can provide multiple capabilities, instead of navigation functions. In conclusion, our system was proven to be a valuable solution for the mobile navigation system. In addition, it is implicated to educate the driver community to be a responsible and alert drivers.

Keywords: *CNN Algorithm, Tiny-YOLOv2, Object Detection, Mobile Navigation System*

1. INTRODUCTION

Malaysia is a developing country in the field of education. This educational centers have produced many successful students in their respective fields of study and every student who has graduated will receive the academic award in an event commonly referred to as a convocation. During convocation, parents and family member will visit the campus, and sometime because of the excitement, they tend to unpracticed the safety driving inside the campus, then, can give hazard to pedestrian and the driver itself. Conventional mobile navigation systems did not equipped with advance driver-assistance. The advance driver-assistance technology usually embedded into car navigation system, which is it depends car model.

Therefore, we aim to develop a mobile navigation system that equipped by two (2) safety additional features, pedestrian and traffic sign detection, which is can give awareness during driving activity. This can ensure this system can be used by many user especially university visitor. This mobile navigation system is named as UTeM navigation system, which will focus only on the main campus of Universiti Teknikal Malaysia Melaka (UTeM), since the main campus is a major place for special occasions such as convocations and other important events.

In this paper, we divide the paper into seven (7) section as follows: Section 2 presents the related work of this research. Section 3 presents problem formulation with research aim. Section 4 presents the proposed system design. Section 5 presents the system implementation. Section 6 presents the evaluation along with result of the proposed algorithm and Section 7 concludes with a discussion and recommendations for further research.

2. RELATED WORK

In this paper, we tend to develop a mobile navigation system with additional safety feature, which are pedestrian detection and traffic sign

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

detection. This safety feature is under the field of Advanced Driver-Assistance Systems (ADASs). Currently, this field have emerged as a prominent component for vehicle safety in modern vehicles. It is important fundamental technology in upcoming self-driving vehicles. The most modern ADASs are vision based, although light detection and range (lidar), radio detection and ranging (radar), and other advanced-sensing technologies are also gaining popularity.

Generally, it can be divided into two (2) types; passive and active. Seat belts, air bags, and padded dashboards are examples of passive safety measures that protect car occupants from injuries after a collision. Because of a continuing consumer demand for safer automobiles, passive safety systems, which have been in continuous development for many decades, have been supplemented with active safety systems, which strive to prevent a crash from occurring in the first place. Active systems are one of the key areas of attention, and they have grown significantly in today's vehicles. Lane keeping, automated braking, and adaptive cruise control are examples of such systems. According to recent World Health Organization statistics, road traffic 9 accidents claim the lives of 1.25 million people each year [1] [2] [3]. Furthermore, with the increasing number of electronic control units and sensor integration, cars now have sufficient processing capability to handle ADAS deployments. Sensors of various types, such as cameras, lidar, radar, and ultrasonic sensors, provide a wide range of ADAS systems. Among these, vision based ADAS, which primarily use cameras as vision sensors, is widely used in today's vehicles. In figure 1 depicts some of the most advanced ADAS capabilities and the sensors needed to accomplish them. It can seen that the ADAS research [4] suggest a taxanomy based on the various type of sensors (see Figure 2).

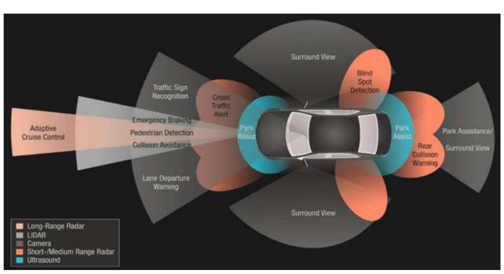

Figure 1: The state of the art ADAS sensor used [4].

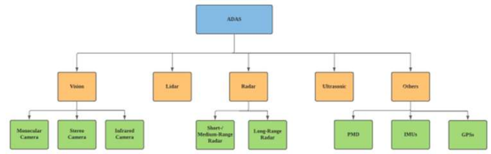

Figure 2: The taxonomy of an ADAS [4].

Based on this taxonomy, it is divided into five different category, which are vision, lidar, radar, ultrasonic and others. Firstly, in vision sensors, cameras are the most frequent type of visual sensor in automobiles. Vision-based ADAS captures images using one or more cameras and an embedded system to detect, analyse, and track various things in them. Cameras may also record information such as colour, contrast, and texture, giving them a distinct advantage over other sensors. For the vision sensors, it has two types of cameras are always used in vision-based ADAS which are monocular and stereo. The monocular camera systems have only one lens. Because these systems only have one image output at any given moment, it has reduced image-processing needs when compared to other camera types and these cameras can be utilized for a variety of purposes, including the detection of obstacles, pedestrians, lanes, and traffic signs [5] [6] [7]. The stereo camera is made up of two or more lenses, each having an image sensor and spaced apart by a specific distance. By matching stereo pairs (pictures from left and right sensors) and using a disparity map to assess the relative depth of a scene, stereo cameras may extract three-dimensional (3-D) information from two or more two-dimensional images. These cameras have substantially higher accuracy than monocular cameras and may be utilized for a range of applications such as traffic sign recognition, lane, pedestrian, and obstacle detection, and distance estimation.

The IR cameras consist of two main which are active IR cameras and Passive IR cameras. Active infrared cameras use a near-IR light source (with wavelengths ranging from 750 to 1,400 nm) integrated into the vehicle to illuminate the scene (which the human eye cannot see) and a regular digital camera sensor to catch the reflected light. Passive infrared cameras employ an IR sensor, with each pixel acting as a temperature sensor capable of detecting thermal radiation emitted by any material. Second category is Lidar, which is used to compute the distance of an item, a laser beam is fired at it and the time it takes for the light to bounce back to the sensor is measured. Lidar is useful in systems that implement automated braking, object detection, collision avoidance, and other functions. Lidars for cars can have a range of up to 60 metres depending on the type of sensor used. Thirdly, the taxanomy of an ADAs is Radar. The radar systems use microwaves to determine an object's speed and distance by monitoring the change in frequency of the reflected wave due to the Doppler effect and radar systems are categorised as short range (0.2–30 m), medium range (30–80 m), or long range (80–200 m) based on their operational distance range [8] [9] [10]. Short-/medium-range radars are used for a variety of purposes, including cross-traffic alerts and blind-spot identification. These systems are frequently found at the vehicle's four corners. Fourthly, for ultrasonic sensor, it employs sound waves to calculate the distance to an object, and mostly used to detect objects that are quite close to the vehicle. Lastly, for other sensor category, a few other sensors are utilised to supplement and improve the functionality of the sensors listed above. Photonic mixer device (PMD) cameras, for example, are made up of an array of smart sensors that allow for quick optical sensing as well as simultaneous demodulation of incoherent light signals [11] [12] [13]. IMUs and GPSs are two systems that aid in the improvement of distance measurements with lidar and radar.

3. PROBLEM FORMULATION AND RESEARCH AIM

Navigation is a common problem for all driver especially university visitor. Unfamiliar place making the driver become careless and unaware, in which give hazard to pedestrian and driver itself. Conventional navigation systems don't have safetly additional feature in which cannot give awareness during driving activity. Thus, in this paper, we aim to solve the issue, by developing a UTeM mobile navigation with safety feature, which pedestrian and traffic sign detection.

4. PROPOSED SYSTEM DESIGN

In this section, we will describe about the functionality of the system. By referring to Figure 3, this system detects person and traffic signs in an image using CNN with Tiny-YOLOv2 architecture. To begin, a dataset with for the two classes is constructed. After the dataset is created, unannotated images of person and traffic sign are manually annotated using the software LabelImg. In this system, the CNN model is used to train on the entire dataset, which included annotations for the person and traffic sign. Then, the system will be tested on real-time video, recorded video, and still images, and finally shows the confidence score as well as the bounding box for each person and traffic sign detected.

I.

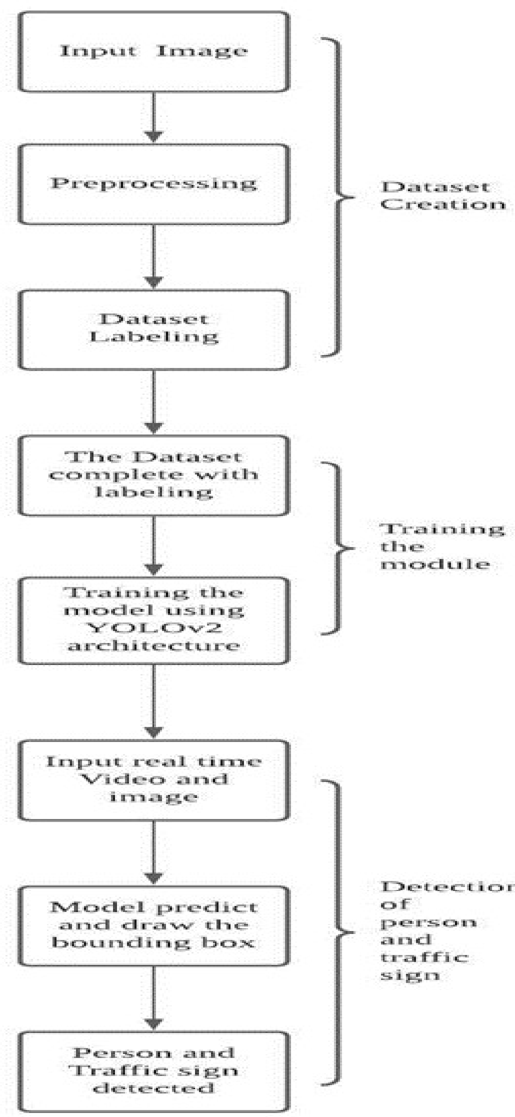

Figure 3: System Architecture

5. SYSTEM IMPLEMENTATION

In this section, we divide the system implementation into two (2) different part which are data preparation and environment setup. For data preparation, the explanation will be focus on on the step of preparation data and type of data. Meanwhile for environment configuration, we will focus on

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

what the package, library or framework that will be used by this system.

A. Data Preparation

In this system, there are 2 classes of images which are person and traffic sign. The person images consist of 500 images and the traffic sign images consist of 500 images. The total of all the image dataset is 1000 images. The data was obtained from version 6 of Google's Open Image Dataset. The images are free to download as the software is open source. Open Image Version 6 makes a significant contribution to the project, as the downloaded images may easily be annotated by downloading their associated annotations. However, not all images of people and traffic signs are taken from Open Image 6. For certain images and traffic sign, the images are taken from the Unsplach website, although the images taken from Unsplach do not readily contain their associated annotations. Manual labelling is carried out with the Labelimg software for these images. By using this software, valuable experience in annotating images is gained.

B. Environment Configuration

In this research, there are several package, library and framework that need be used, which are Flutter, Mapbox, TensorFlow, and Tiny-YOLOv2. These very crucial because it is compulsory before the system can e executed. Firstly, Flutter is an open source mobile application development framework. It is used to create apps for Android and iOS, as well as the primary approach for developing programmes for Google Fuchsia. Flutter apps are created in Dart language and utilise many of the language's more advanced features. Using flutter, it needs to add and import necessary the packages flutter that related to the model that used in this project. In figure 4, it shows the packages that used in this project.

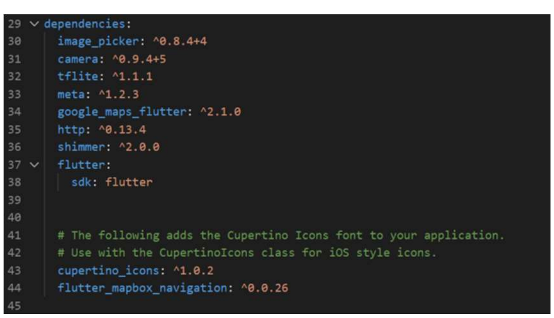

Figure 4: The packages used in this project

To implement the CNN model that have created, it needs to import all the necessary packages to run the model. Figure 5 show the packages that have imported into code.

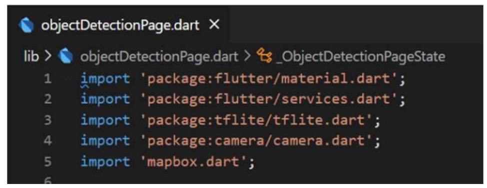

Figure 5: The packages for object detection

For the part of map navigating, it used Mapbox. Mapbox is the mapbox maps that support in flutter, and it need to add the mapbox access token. Mapbox uses access tokens to associate API requests with user account. The Figure 6 below show the mapbox access token that used in this project.

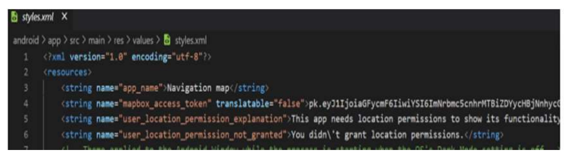

Figure 6: The mapbox access token that used in this project.

For the part of object detection, it uses a Tensorflow lite file format (.tflite). Tensorflow lite file is a set of tools that enables on device machine learning that helps to run the models on mobile devices. It supports the machine learning task such as object detection and image classification. To get the detection, it needs to convert the Tiny-YOLOv2 weights.file into Protubuf file format (.pb). Next, the protubuf file format convert to Tensorflow lite file format (.tflite). In Figure 7, it shows the TinyYOLOv2 weights file format. The Figure 8 show the protubuf file format and tflite file format. The Figure 9 show the tflite file format that used in this system for the object detection.

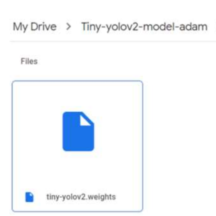

Figure 7: The Tiny-YOLOv2 weights file format.

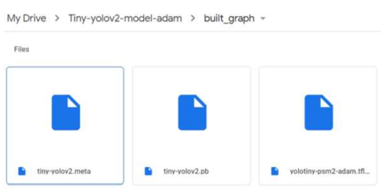

Figure 8: The Tiny-YOLOv2 protubuf file format and tflite file format.

```
loadModel() async {
  Tflite.close();
  try {
    String res = await Tflite.loadModel(
      model: "assets/yolotiny-psm2-adam.tflite",
      labels: "assets/labels.txt",
      // useGpuDelegate: true,
    );
    print(res);
  } on PlatformException {
    print('Failed to load model.');
  }
}
```

Figure 9: The tflite file format that implemented in flutter for the object detection

The proposed system uses CNN algorithm with Tiny-YOLOv2 architecture. CNN is a deep learning algorithm that can take in an image as input, assign weight to numerous objects in the image, and distinguish between them. It also minimizes the images into a format that is easier to process while saving important features for a good prediction. CNN have several architectures. For this system, Tiny-YOLOv2 is used as a CNN architecture. YOLOv2 is an object detection model that can perform well on real-time object detection. It requires only one evaluation to predict bounding boxes and class probabilities from complete images. It also predicts all bounding boxes for an image across all classes at the same time. The Tiny Darknet is a classification model that is used as the backbone for Tiny-YOLOv2. The layers of Tiny Darknet consist of 15 layers which are 9 convolutional layers and 6 maxpooling layers. The convolutional layers of the Tiny-Darknet model use convolutional weights that are pre-trained on the ImageNet dataset. This model is optimized by using Adam with initial learning rate of 0.00001. Before building and compiling the model, important libraries such TensorFlow and OpenCV first need to be imported. The datasets will be processed using image augmentation to rotate, rescale, and modify the images so that the model can learn the pattern quickly and accurately. After all of the necessary requirements had been fulfilled, the model was trained for 300 epochs and then saved. For future use, this model will be implemented on the native application.

6. EVALUATION AND RESULT

In this section, we will discuss about the performance evaluation of the system. Initially, we will start by open our system (see Figure 10) and do the navigation activity while driving car. The location of this experiment is in within UTeM main campus. When doing the navigation activity, we also activate the pedestrian and traffic sign detection. As we can see on the smart phone screen, the system can do navigation (see Figure 12 and Figure 13) and also able to detect the person and traffic sign in real time (see Figure 11).

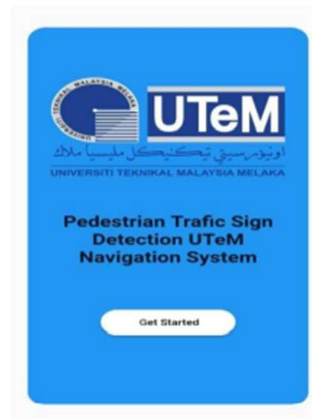

Figure 10: The homepage for this system

Figure 11: The camera object detection page and location page for this system

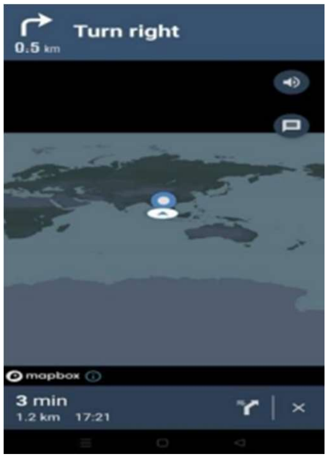

Figure 12: The page of map navigation

Figure 13: The page of map navigation

After did the navigation activity while driving, then we can obtain the accuracy and mean average precision of the model by using two different optimizers, Adam and RMSprop. The model has been trained using the same CNN algorithm, data, and learning rate. The goal of this analysis is to compare the best optimizer to use for this model. An optimizer is an algorithm that is used to adjust the properties of a neural network, such as its weights and learning rate, in ways that minimize the model's loss and provide the best accurate result feasible.

For Adam optimizer, we can see the result of mean average precision (see Figure 14) for the model that using Adam optimizer is 90.44%, with the average precision for person is 0.91 and average precision for traffic sign is 0.90. As final accuracy (see Figure 15), the system can achieve as 0.9175. Meanwhile, for RMSprop optimizer, we can see the result of mean average precision (see Figure 16) for the model that using Adam optimizer is 78.19%, with the average precision for person is 0.75 and average precision for traffic sign is 0.81. As final accuracy (see Figure 17), the system can achieve as 0.793.

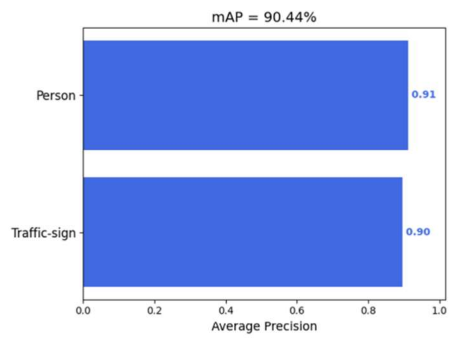

Figure 14: Result mAP for model that using Adam optimizer

```
undetected: 0     True positives: 1    False positives: 0
undetected: 0     True positives: 1    False positives: 0
undetected: 0     True positives: 1    False positives: 0
undetected: 0     True positives: 1    False positives: 0
Final Accuracy 0.9175
```

Figure 15: Result final accuracy for model that using Adam optimizer

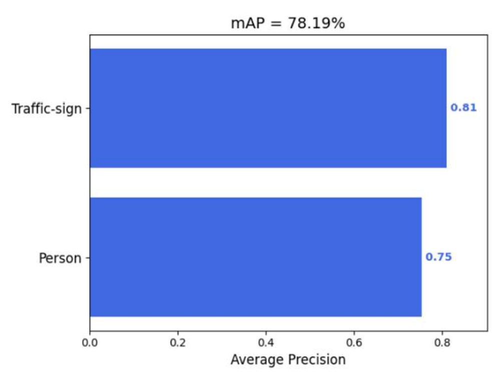

Figure 16: Result mAP for model that using RMSprop optimizer.

```
undetected: 0     True positives: 1     False posit
undetected: 0     True positives: 1     False posit
Final Accuracy 0.793
```

Figure 17: Result final accuracy for model that using RMSprop optimizer.

From this result comparison, we can see clearly that the model with Adam optimizer give better result rather than model with RMSprop optimizer based on the result mean Average Precision, average precision for each class and the final accuracy. This is due to Root Mean Square Propagation (RMS-prop) having the learning rate is the gradient's exponential average rather than the cumulative sum of squared gradients. The RMS-Prop is also slow learner and have the longest training time.

7. CONCLUSION AND FUTURE WORK

As conclusion, we already develop a mobile navigation system (so called as UTeM navigation system) that have additional safety feature, pedestrian detection with traffic sign detection. This development is to solve common problem that faced by all driver especially university visitor, by giving awareness to driver itself. Conventional navigation system did not equipped with safety feature, thus can give hazard to pedestrian and driver itself. In this system, the CNN algorithm and the architecture of Tiny-YOLOv2 was used to detect traffic sign and pedestrian. As our finding, our proposed system showing that the mean average precision for both classes can achieve as 90.44%, when it implemented in conventional smart phone. This is showing that our system have good capability when it implemented with mobile navigation system. As addition, our system will be proven to be a valuable solution for mobile navigation system. For future work, we plan to try different architecture for person detection and traffic sign detection.

8. ACKNOWLEDGEMENT

This study was funded by UTeM Short Term Research Grant which is titled as “A Rapid Flood Emergency Responce Solution During Flood Based On Human Activity Recognition And Classification” (No: PJP/2020/FTMK/PP/S01801) . The author also thanks to all the research member his or her astute comments during the preparation of the rough draft of this paper

REFERENCES:

[1] Q. Yuan and J. Wang, “Goods movement, road safety, and spatial inequity: Evaluating freight-related crashes in low-income or minority neighborhoods,” *J. Transp. Geogr.*, vol. 96, p. 103186, Oct. 2021, doi: 10.1016/j.jtrangeo.2021.103186.

[2] L. Zheng, T. Sayed, and F. Mannering, “Modeling traffic conflicts for use in road safety analysis: A review of analytic methods and future directions,” *Anal. Methods Accid. Res.*, vol. 29, p. 100142, Mar. 2021, doi: 10.1016/j.amar.2020.100142.

[3] W. K. M. Alhajyaseen *et al.*, “Road safety status during COVID-19 pandemic: exploring public and road safety expert’s opinions,” *Int. J. Inj. Contr. Saf. Promot.*, vol. 0, no. 0, pp. 1–17, Aug. 2021, doi: 10.1080/17457300.2021.1962915.

[4] V. K. Kukkala, J. Tunnell, S. Pasricha, and T. Bradley, “Advanced driver-assistance systems: A path toward autonomous vehicles,” *IEEE Consum. Electron. Mag.*, vol. 7, no. 5, pp. 18–25, 2018.

[5] Q. Lu, H. Zhou, Z. Li, X. Ju, S. Tan, and J. Duan, “Calibration of five-axis motion platform based on monocular vision,” *Int. J. Adv. Manuf. Technol.*, vol. 118, no. 9, pp. 3487–3496, Feb. 2022, doi: 10.1007/s00170-021-07402-x.

[6] J. Liu, R. Zhang, and S. Hou, “Inter-vehicle distance estimation considering camera attitude angles based on monocular vision,” *Proc. Inst. Mech. Eng. Part J. Automob. Eng.*, vol. 235, no. 2–3, pp. 894–900, 2021.

[7] J. Zhu, F. Shi, and J. Li, “Advanced Driver Assistance System based on Machine Vision,” in *2021 IEEE 4th Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC)*, Jun. 2021, vol. 4, pp. 2026–2030. doi: 10.1109/IMCEC51613.2021.9482067.

[8] N. Susithra, G. Santhanamari, M. Deepa, P. Reba, K. C. Ramya, and L. Garg, “Deep Learning-Based Activity Monitoring for Smart Environment Using Radar,” in *Challenges and Solutions for Sustainable Smart City Development*, R. Maheswar, M. Balasaraswathi, R. Rastogi, A. Sampathkumar, and G. R. Kanagachidambaresan, Eds. Cham: Springer International Publishing, 2021, pp. 91–123. doi: 10.1007/978-3-030-70183-3_5.

[9] H. Yang, Q. Wang, H. Li, F. Fang, C. E. Montenegro-Marin, and S. N. Kadry, "Maritime moving object localization and detection using global navigation smart radar system," *Soft Comput.*, vol. 25, no. 18, pp. 11965–11974, Sep. 2021, doi: 10.1007/s00500-021-05625-4.

[10] V. Kovregin and G. Kovregin, "Adaptive-robust Methods for Detecting, Capturing and Tracking Hovering, Low- and High-speed Objects in Integrated Radar-Inertial Systems with Quasi-continuous Radiation," in *2021 28th Saint Petersburg International Conference on Integrated Navigation Systems (ICINS)*, May 2021, pp. 1–4. doi: 10.23919/ICINS43216.2021.9470845.

[11] K. Klionovska, J. Ventura, H. Benninghoff, and F. Huber, "Close Range Tracking of an Uncooperative Target in a Sequence of Photonic Mixer Device (PMD) Images," *Robotics*, vol. 7, no. 1, Art. no. 1, Mar. 2018, doi: 10.3390/robotics7010005.

[12] P. Kienitz, A. Bablich, R. Bornemann, and P. H. Bolívar, "Photonic mixer device (PMD) based on graphene for high-resolution 3D sensors," in *2D Photonic Materials and Devices IV*, 2021, vol. 11688, p. 116880S.

[13] A. Satou *et al.*, "Unitraveling-Carrier-Photodiode-Integrated High-Electron-Mobility Transistor for Photonic Double-Mixing," *J. Light. Technol.*, vol. 39, no. 10, pp. 3341–3349, 2021.