Received: 28 February 2023 | *Revised: 4 April 2023* | *Accepted: 4 April 2023*
DOI: 10.37917/ijeee.19.2.8 *Vol. 19 | Issue 2 | December 2023*

Open Access

Iraqi Journal for Electrical and Electronic Engineering
Original Article

IJEEE University of Basrah College of Engineering

Feature Deep Learning Extraction Approach for Object Detection in Self-Driving Cars

Namareq Odey *, Ali Marhoon
[1]Electrical Engineering Department, University of Basrah, Basrah, Iraq

Correspondance
* Namareq Odey
Electrical Engineering department
University of Basrah, Basrah, Iraq
Email: namareqalmotter22@gmail.com

Abstract
Self-driving cars are a fundamental research subject in recent years; the ultimate goal is to completely exchange the human driver with automated systems. On the other hand, deep learning techniques have revealed performance and effectiveness in several areas. The strength of self-driving cars has been deeply investigated in many areas including object detection, localization as well, and activity recognition. This paper provides an approach to deep learning; which combines the benefits of both convolutional neural network CNN together with Dense technique. This approach learns based on features extracted from the feature extraction technique which is linear discriminant analysis LDA combined with feature expansion techniques namely: standard deviation, min, max, mod, variance and mean. The presented approach has proven its success in both testing and training data and achieving 100% accuracy in both terms.

Keywords
SELF-DRIVING CARS, FEATURE EXPANSION, DEEP LEARNING, CNN, LDA.

I. Introduction

Autonomous Vehicle(AV) performs an important field both in modern mechanical and electrical engineering as well as intelligent transportation systems. The field of self–driving cars is evolving very fast, it utilizes a combination of sensors, actuators, machine learning systems, and complex and powerful algorithms to implement software and travel between destinations without human interference.

On the other hand, deep learning is an improved subdomain of machine learning. It enhances the modeling of complex relationships and concepts using multiple levels of representation. Supervised and unsupervised learning algorithms are used to construct successively higher levels of abstraction, defined using the output features from lower levels.

However, object detection using deep learning is at the core of a great number of computer vision applications like face detection [**?**], video surveillance [1], optical character recognition [2], and object counting/tracking [3],[4].

More specifically, in self-driving cars [5], the ultimate task of the perception system of a vehicle computer is to reveal nearby objects simultaneously with making the optimal decision like path planning as well as collision avoidance. Autonomous vehicles have gained explosive improvement during the past decade [6].

Self-driving cars are under the spotlight and gained tremendous improvement throughout the past decade [7], despite the technological improvement, autonomous vehicle systems are still far away from being trustworthy and reliable. Several accidents in self-driving cars are caused by misclassification or not recognized objects [8]; therefore, increasing the capability of object detection in these systems is a very high priority.

Recent studies in deep learning have boosted the performance of object detection algorithms to a higher level [9],[10]. This paper introduces an approach of deep learning mimics the human brain in case of learning based on features instead of images, to be used in Pedestrians, Cars, bakers, trucks and traffic lights recognition and identification aiming to develop autonomous driving technologies.

II. Related work

In this section, the main previous works that address the object detection process using deep learning models are covered. Ayegül Uçar et al. [11] suggest two Convolutional Neural Networks (CNNs) deep models with different layers and use linear Support Vector Machine (SVM) classifiers in the training features step. This work uses Caltech 101 and Caltech pedestrian detection datasets. The average accuracy rate in this work is 92.80+0.43%.Kai Janet et al. [12] utilize faster region Convolutional Neural Networks (faster RCNN) with Residual Neural Networks (ResNet 101) as adeep model. Use it to recognize the scratches points on the wheel hub even if the image data has a complicated background, in addition, the model can recognize variance types of wheel hub defects as well as determine the class and position of the defective area. This work establishes a database consisting of defect area images and achieves an average accuracy rate of 86.3%.

Stefan Schneider et al. [13] compare two deep learning models used for object detection namely, Faster Region-Convolutional Neural Network together with You-Only-Look-Once v2.0 (YOLOv2) in the Serengeti dataset, the results prove that the first method explicitly promises accuracy of 93%, where YOLO fails to perform the same accuracy and achieves 76.7%.

Chandan G et al. [[14] develop a MobileNets-based technique with a Single-Shot Detector (SSD) technique for tracking as well as detection of the camera-based dataset in a python environment. These techniques include detecting the region of interest of an object from a given image class and achieving an average accuracy rate of 99.0%. CNN is used for feature extractions.

Xiangmo Zhao et al. [15] propose a model that uses the complementarity of the camera and 3D LIDAR and camera data to recognize multiple objects around an autonomous car. The average accuracy rates of the accuracy of the proposed method reached 89.04% and 78.18%, respectively in the detection of vehicles and pedestrians at a moderate level of difficulty.

Muhammad Shakeel et al [16] propose an accurate drowsiness detection methodology based on object detection using Convolutional Neural Networks. Which achieves an accuracy of 84% in their own new, annotated Drowsy dataset.

Muhammad Yahya et al. [17] provide an improvement in object detection using LiDAR-collected data by using different algorithms like YOLOv2, and Interactive Multi-Mode (IMM). The average accuracy rates in the newly collected data set are 71.43% and 57.14% respectively.

YANFEN LI et al [18] suggest an object recognition as well as detection theme for autonomous driving. The suggested theme is capable to detect ten kinds of objects; the proposed model is based on YOLOv4 and achieves an average accuracy rate of 52.7% in BDD100k and three collected datasets.

In this work feature extraction and feature expansion of the udacity images dataset are performed to assist deep model classification the features are extracted by a linear discriminant analysis and the feature expansion contribute six operations namely: standard deviation, min, max, mod, variance, and mean classified as five classes which are bikers, cars, pedestrians, traffic lights and trucks.

III. Methodology

the proposed method is based on five main stages, as explained in Figure-1: the first stage, is the data collection stage where we use data from the Udacity Self Driving Car Dataset dataset. Our data is distributed in five classes,namely: bakers, cars, pedestrians, traffic lights, and trucks, each class has a different number of files.

The second stage is an image augmentation step, the expected result from this stage is more data where we will get images for objects with different angles and positions as well as increase the number of images in each class so that the number of images in the different classes will be quit equally to overcome the over-fitting issues.

On the other hand, the third stage, the pre-processing stage in which we get a more enhanced dataset where we manipulate the smoothing and blurring of our images to make the detection of the desired objects easier and possible under all conditions of noise and lightness issues so that the dataset will be more appropriate to the next feature extraction stage whereas, the fourth stage is the feature extraction and expansion stage where we extract features from our image dataset, then feed these features to our hybrid model. The final stage is our proposed supervised deep learning model which employs the benefits of CNN and Dense techniques to implement a hybrid model that mimics the human brain in case of learning based on features instead of images used to achieve object detection dedicated to self-driving cars.

A. The Dataset

This study was evaluated based on features extracted from udacity dataset, which is dedicated mainly to data to be provided in autonomous cars. Primarily 110 MB dataset was downloaded, then many processes are accomplished to enhance and improve our dataset before turning them into features, through the data augmentation and image processing steps.

B. Data Augmentation

Data augmentation procedures apply different operations to the images data to provide more images for objects with different angles and positions as well as increase the number of images to overcome the issue of over fitting, the most meaningful principles adopted in data augmentation are flipping and rotation [19], which is used in our model, where flipping produces a mirror copy for an original image with bother vertical or horizontal axes. Flipping about the horizontal axis is preferred due to the upper and lower parts of an image have not changed.

Where in case of rotation the images are rotated right or left around 10o. The safety of the rotation augmentation technique is determined by the rotation angle, where the incensement in the rotation degree may cause the image label to be not preserved.

C. Pre-processing Stage

Image pre-processing is a multidimensional process, which is utilized to get a more enhanced dataset where we manipulate the smoothing and blurring of our images to make the detection of the desired objects easier and capable under conditions of noise and lighting issues. Some of the pre-processing techniques are used in our system, first, our images data are converted to grey-scale images then operations like histogram equalization, Gaussian blur, and resizing are applied to our data so that they will get prepared for the next stage of our system, these techniques are briefly explained below [20],[21],[22]:

1) Grey Scale Conversion

The converting of an image from the colour Red, Green, and Blue (RGB) space to grey-scale space is a process of mapping from a higher dimensional vector space into a lower dimensional vector space Many linear and nonlinear methods are utilized for the grey space conversion process. One of the basic methods is the weighted or luminosity method it forms a weighted average to account for human perception which means that the contribution of the red colour has to be decreased and increase the contribution of the green colour and put blue colour contribution in between these two.

$$GrayScaleImage = ((0.3 \times R) + (0.59 \times G) + (0.11 \times D)) \quad (1)$$

2) Histogram Equalization

Histogram equalization can be considered a method of contrast enhancement in image processing by means of an image's histogram. It is essential to function in global contrast augmentation of the image and intensity adjustment to be better distributed in the histogram, which can allow lower local areas of contrast to profit from a higher contrast.

More particularly, the histogram is a cumulative distribution function, which is important in computing histogram equalization as shown in the following equation.

$$C_{df} = \sum_{i=1}^{x} h(i) \quad (2)$$

Where X represents the grey value and h illustrates the image's histogram.

$$T[pixel] = round((\frac{cdf(x) - cdf(x)_{min}}{E * f - cdf(x)_{min}})) * (L-1) \quad (3)$$

$cdf(x)_m in$ is the minimum value of the cumulative distribution function.

E*f Columns and rows number of images.

L: Grey levels used =256.

3) GuassianBlur

GuassianBlur is a linear low-pass filter used for blurring, smoothing, and eliminating noise in images. The following function can be used to determine the GuassianBlur.

$$G(x,y) = \frac{1}{(2\pi\partial^2)e^{\frac{x^2+y^2}{2\sigma^2}}} \quad (4)$$

Where x is the distance from the origin in the horizontal axis.

y is the distance from the origin in the vertical axis.

σ is the standard deviation of the Gaussian distribution.

4) Image Resize

The resizing operation has a significant role to reduce as well as enhance the image size. Image interpolation is achieved in two different ways: by down-sampling or up-sampling the image. Choosing a precise interpolation method is essential in various applications.

D. Feature Extraction And Expansion

Our proposed model is a feature approach, the features are extracted from our data set by utilizing the LDA feature extraction method, then to improve and increase the number of probabilities to our model the produced features are expanded by some expansion techniques which are: Mean Min, Max, Standard Deviation and Variance.

The number of features extracted from one image by LDA is (the number of classes -1), in our approach equals 4 since we have 5 classes, then the features are expanded by six feature expansion techniques so the total features extracted from one image is 24. So the total number of features extracted from our dataset is 24(which is the total number of features extracted from one image) * 2687 (which is the total number of images data set after image augmentation and preprocessing steps).

1) Linear Discriminant Analysis (LDA) feature extraction

Although the linear discriminant analysis LDA technique mainly projects the high dimensional data into lower dimensional space, in this study it is used as a feature extraction technique since it is applied directly to images. LDA intends to minimize the within-class distance and maximize the between-classes distance in the dimensionally reduced space

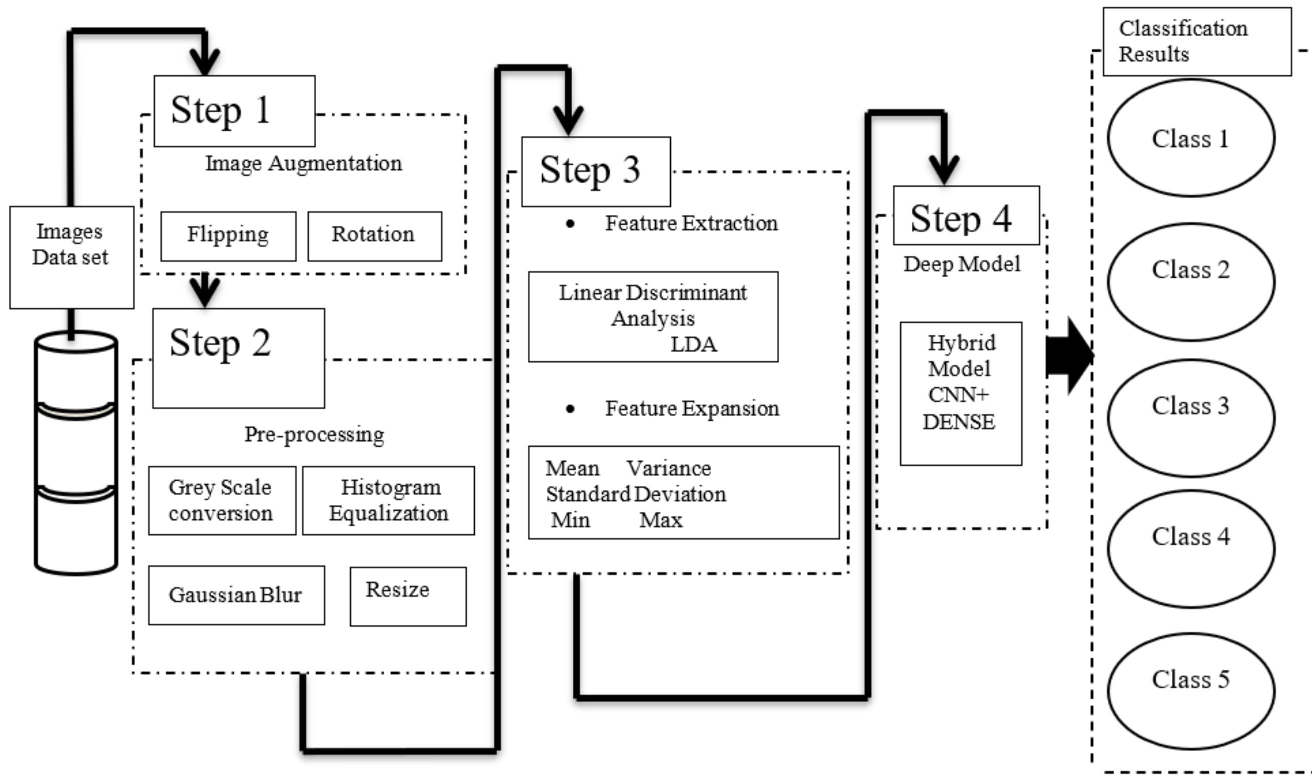

Fig. 1. Framework of the proposed method

[23],[24],[25] the LDA can be computed using the following equations:

$$trace((X^T S_W X)^{-1}(X^T S_b X)) \tag{5}$$

$$S_b = \frac{1}{n}\sum_{i}^{m} K_i(c_i - c)(c_i - c)^T \tag{6}$$

$$S_w = \frac{1}{m}\sum_{i=1}^{m}\sum_{x \epsilon x_i}(x - c_i)(x - c_i)^T \tag{7}$$

Where

X is the sample of our data.

S_W is within the class matrix.

S_b is the between-class matrix.

c is the number of distinct classes.

2) Features Expansion Techniques

Features extracted from LDA are expanded using statistical features techniques to maximize the number of deep model probabilities so that our deep model will be more efficient and robust. Statistical features used in this stage are Mean, Max, Min, standard deviation, variance, and Mod operations [26], these operations are applied as a sliding window to the LDA features using C++ functions.

IV. DEEP LEARNING MODEL

Our deep model is a (37) layer network compromising of (11) deep layers formed by one-direction convolutional layers, (6) fully connected layers formed by dense layers, and the remaining layers Consist of 9 Max pooling layers, 1 normalization layer represented by flatten layer, 10 leaky rectified linear units (leaky ReLU) activation layers and one dropout layer.

The (720,984) features are divided into receptive fields that feed into a convolutional layer. The network has an input size of 24 features, all the layers are piled up on each other or arranged one after the other. The convolution layers are based on filters of different sizes which are 16, 32, 64, 128, 256, 512, 1024, 1024, 512, 512, and 50 respectively with kernel size 3, the stride of 1, with the same padding. The Max pooling layers have stride=1, size=2, and the same padding.

Whereas the linear collectors or Dense layers have different kernel sizes of 128,512, 1024, 32, 16, and 5 respectively and different activation functions namely the linear and soft-

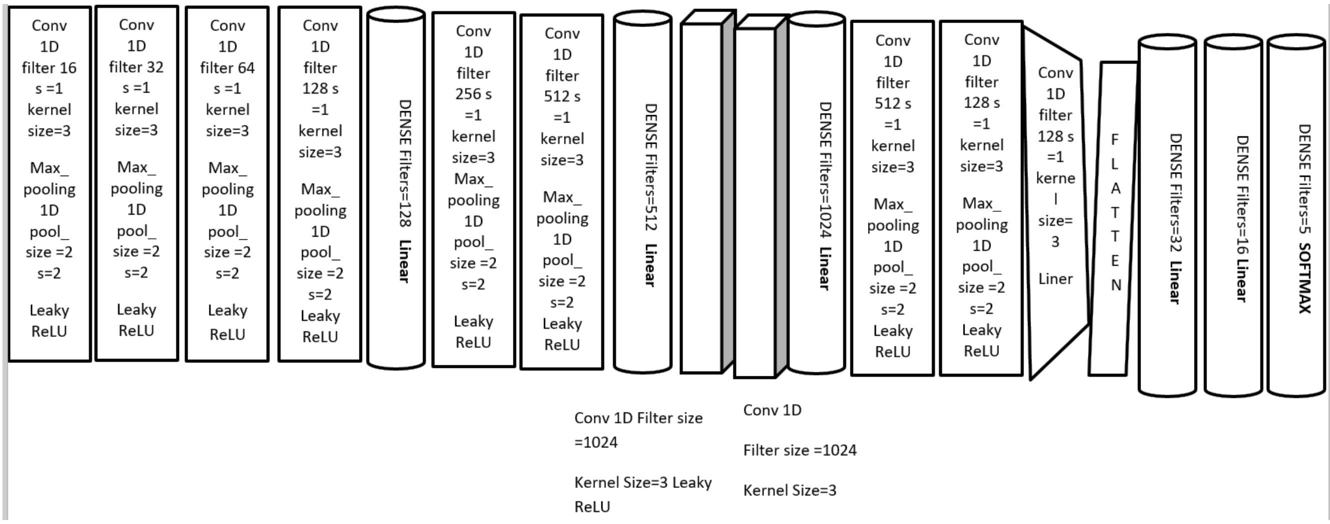

Fig. 2. Flow chart of our deep model

max. Our model also contains one dropout of 0.6 and one normalization layer represented by flatten. Adam optimizer is used to update our deep network weights; our model is summarized in Figure-2.

V. Results

In this study, (720,984) features were extracted from (2687) image data which we get after some image augmentations and preprocessing operations of the original (6,499) images downloaded from udacity website.

Our deep model has a number of parameters (3,305,095) and it is trained in 100 epochs, the main advantage of the proposed model is to reduce computational time as the training time for one epoch is between 15 to 10 seconds as obtained in Figure-3.

```
Train on 2687 samples, validate on 2687 samples
Epoch 1/100
2023-02-21 03:23:31.112472: I T:\src\github\tensorflow\tensorflow\core\platform\cpu_feature_guard.cc:140] Your CPU supports instr
2687/2687 [==============================] - 16s 6ms/step - loss: 1.1782 - acc: 0.5579 - val_loss: 0.7774 - val_acc: 0.7175
Epoch 2/100
2687/2687 [==============================] - 17s 6ms/step - loss: 0.6162 - acc: 0.7797 - val_loss: 0.4645 - val_acc: 0.8478
Epoch 3/100
2687/2687 [==============================] - 16s 6ms/step - loss: 0.3858 - acc: 0.8683 - val_loss: 0.3105 - val_acc: 0.8984
Epoch 4/100
2687/2687 [==============================] - 16s 6ms/step - loss: 0.2967 - acc: 0.8991 - val_loss: 0.2930 - val_acc: 0.8988
Epoch 5/100
2687/2687 [==============================] - 15s 6ms/step - loss: 0.2764 - acc: 0.9066 - val_loss: 0.2753 - val_acc: 0.9025
Epoch 6/100
2687/2687 [==============================] - 20s 8ms/step - loss: 0.2633 - acc: 0.9129 - val_loss: 0.2363 - val_acc: 0.9207
Epoch 7/100
1600/2687 [=================>............] - ETA: 5s - loss: 0.2498 - acc: 0.9137
```

Fig. 3. Computational time for epoch training

Our model learns based on features, 70% of our features were used for training the model and the remaining 30% of the features were used for testing our model. Our model was evaluated in terms of accuracy, Recall, Precision as well as F1 score and shows promising results of 100% for accuracy in both training and testing phases, and 1 for the rest parameters for each class as shown in Figure -4.

```
2687/2687 [==============================] - 13s 5ms/step
              precision    recall  f1-score   support

           0       1.00      1.00      1.00       513
           1       1.00      1.00      1.00       613
           2       1.00      1.00      1.00       455
           3       1.00      1.00      1.00       595
           4       1.00      1.00      1.00       511

    accuracy                           1.00      2687
   macro avg       1.00      1.00      1.00      2687
weighted avg       1.00      1.00      1.00      2687

1.0 1.0 1.0
```

Fig. 4. Values of evaluating parameters

Furthermore, our model is also executed in real-time mode, although the low-resolution camera and noisy and uncertain YouTube data used it shows promising results, samples of samples of the exactions results are shown in figure-5(a-f).

A comparison between our model and some models provided by some previous related works in terms of Accuracy, datasets and procedures has been shown in Table-I.

VI. Conclusion

In this work, a comprehensive deep learning model has been proposed to be used in baker, cars, pedestrians, traffic lights, and trucks recognition and identification aiming to develop autonomous driving technologies. our approach shows promising results in terms of accuracy, reducing training time, and using less computational resources The given model is a deep 37-layer network architecture, which has been adopted after many extensive trials. The learning process was accom-

(a) (b) (c) (d) (e) (f)

Fig. 5. Detection of the five classes by our model under different conditions

plished depending on features extracted from udacity data set images, which has been divided to use for training and testing data set. The adopted approach has been proven its successful in correctly identifying 100% of testing data set.

Conflict of Interest

The authors have no conflict of relevant interest to this article.

References

[1] H. Liang, X. Sun, Y. Sun, and Y. Gao, "Text feature extraction based on deep learning: a review," *EURASIP journal on wireless communications and networking*, vol. 2017, no. 1, pp. 1–12, 2017.

[2] M. Masmoudi, H. Friji, H. Ghazzai, and Y. Massoud, "A reinforcement learning framework for video frame-based autonomous car-following," *IEEE Open Journal of Intelligent Transportation Systems*, vol. 2, pp. 111–127, 2021.

[3] B. Alvey, D. T. Anderson, A. Buck, M. Deardorff, G. Scott, and J. M. Keller, "Simulated photorealistic deep learning framework and workflows to accelerate computer vision and unmanned aerial vehicle research," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3889–3898, 2021.

[4] V. Thanh Dat, P. Quang Huy, N. Dinh Tra, T. Hai Anh, B. Ngoc Anh, and N. Tung Son, "A deep learning based implementation for self-driving car," in *2021 10th Inter-*

TABLE I.
Comparison between Our Model with Other Models

	Year	Procedure	Datasets	Accuracy
Ayegül Uçar et al	2016	CNN with different layers	Caltech Caltech pedestrians defection dataset	92.80+-0.43%
Kai Hanet et al	2017	Faster_RCNN ResNet101	Established a database consisting of defect area images	86.3%
Stefan Schneider et al	2018	Faster_RCNN YOLO V2.0	Reconyx camera trap self labeled gold standard snapshot Serengeti dataset	93.0% 76.7%
Chandan G et al	2018	Mobile Nets with SSD	Camera	99.0%
Xiangmo Zhao et al	2019	3d lider CNN Camera CNN	Kitti dataset	89.04% 78.18%
Muhammad Faique Shakeel et al	2019	Novel deep based on CNN	Custom dataset	84%
Muhammad Azri Yahya et al	2020	Lider YOLO v2.0 SSD	Collected Data	71.43% 57.14%
YANFEN LI et al	2020	Modified YOLOv4	BDD100k Three collected datasets	52.7%
Our model	2022	Proposed hybrid model	Our features dataset	100%

national Conference on Software and Computer Applications, pp. 279–287, 2021.

[5] X. Ye and Q. Zhu, "Class-incremental learning based on feature extraction of cnn with optimized softmax and one-class classifiers," *IEEE Acces*, vol. 7, pp. 42024–42031, 2019.

[6] M. Maggipinto, A. Beghi, S. McLoone, and G. A. Susto, "Deepvm: A deep learning-based approach with automatic feature extraction for 2d input data virtual metrology," *Journal of Process Control*, vol. 84, pp. 24–34, 2019.

[7] X. Li, W. Zhang, and Q. Ding, "Deep learning-based remaining useful life estimation of bearings using multiscale feature extraction," *Reliability engineering & system safety*, vol. 182, pp. 208–218, 2019.

[8] Y. Xianjia, S. Salimpour, J. P. Queralta, and T. Westerlund, "Analyzing general-purpose deep-learning detection and segmentation models with images from a lidar as a camera sensor," *arXiv preprint arXiv:2203.04064*, 2022.

[9] M. Abdulla and A. Marhoon, "Agriculture based on internet of things and deep learning. iraqi journal for electrical and electronic engineering 18.2 (2022): 1-8.kim, cheol-jin, et al.end-to-end deep learning-based autonomous driving control for high-speed environment.," *The Journal of Supercomputing*, vol. 78, no. 2, pp. 1961–1982, 2022.

[10] H. Hakim and A. Fadhil, "Survey: Convolution neural networks in object detection.," *Journal of Physics: Conference Series.*, vol. 1804, no. 1, 2021.

[11] A. Uçar, Y. Demir, and C. Güzeliş, "Moving towards in object recognition with deep learning for autonomous driving applications," in *2016 International Symposium on INnovations in Intelligent SysTems and Applications (INISTA)*, pp. 1–5, IEEE, 2016.

[12] K. Han, M. Sun, X. Zhou, G. Zhang, H. Dang, and Z. Liu, "A new method in wheel hub surface defect detection: Object detection algorithm based on deep learning," in *2017 International Conference on Advanced Mechatronic Systems (ICAMechS)*, pp. 335–338, IEEE, 2017.

[13] S. Schneider, G. W. Taylor, and S. Kremer, "Deep learning object detection methods for ecological camera trap data," in *2018 15th Conference on computer and robot vision (CRV)*, pp. 321–328, IEEE, 2018.

[14] G. Chandan, A. Jain, and H. Jain, "Real time object detection and tracking using deep learning and opencv," *2018 International Conference on inventive research in computing applications (ICIRCA). IEEE*, 2018.

[15] X. Zhao, P. Sun, Z. Xu, H. Min, and H. Yu, "Fusion of 3d lidar and camera data for object detection in autonomous vehicle applications," *IEEE Sensors Journal*, vol. 20, no. 9, pp. 4901–4913, 2020.

[16] M. F. Shakeel, N. A. Bajwa, A. M. Anwaar, A. Sohail, and A. Khan, "Detecting driver drowsiness in real time through deep learning based object detection," in *International work-conference on artificial neural networks*, pp. 283–296, Springer, 2019.

[17] M. A. Yahya, S. Abdul-Rahman, and S. Mutalib, "Object detection for autonomous vehicle with lidar using deep learning," *2020 IEEE 10th International conference on system engineering and technology (ICSET), IEEE*, 2020.

[18] Y. Li, H. Wang, L. M. Dang, T. N. Nguyen, D. Han, A. Lee, I. Jang, and H. Moon, "A deep learning-based hybrid framework for object detection and recognition in autonomous driving," *IEEE Access*, vol. 8, pp. 194228–194239, 2020.

[19] P. Oza, P. Sharma, S. Patel, F. Adedoyin, and A. Bruno, "Image augmentation techniques for mammogram analysis," *Journal of Imaging*, vol. 8, no. 5, p. 141, 2022.

[20] S. Perumal and T. Velmurugan, "Preprocessing by contrast enhancement techniques for medical images," *International Journal of Pure and Applied Mathematics*, vol. 118, no. 18, pp. 3681–3688, 2018.

[21] V. M. Mohan, R. Kanaka Durga, S. Devathi, and K. Srujan Raju, "Image processing representation using binary image; grayscale, color image, and histogram," in *Proceedings of the Second International Conference on Computer and Communication Technologies: IC3T 2015, Volume 3*, pp. 353–361, Springer, 2016.

[22] L. M. Rancis and N. Sreenath., "Pre-processing techniques for detection of blurred images," *Proceedings of International Conference on Computational Intelligence and Data Engineering, Springer, Singapore*, 2019.

[23] Y. Wang and Y. Chen, "A new feature extraction algorithm based on fisher linear discriminant analysis," *3rd International Conference on Control, Automation and Robotics (ICCAR). IEEE*, 2017.

[24] C. Castaldello, A. Gubert, F. Galvanin, A. Casonato, R. Padrini, M. Barolo, and F. Bezzo, "A model-based support for diagnosing von willebrand disease," in *Computer Aided Chemical Engineering*, vol. 40, pp. 2779–2784, Elsevier, 2017.

[25] A. O. Salau and S. Jain, "Feature extraction: a survey of the types, techniques, applications," *2019 International Conference on Signal Processing and Communication (ICSC), IEEE*, 2019.

[26] R. Zebari, A. Abdulazeez, D. Zeebaree, D. Zebari, and J. Saeed, "A comprehensive review of dimensionality reduction techniques for feature selection and feature extraction," *Journal of Applied Science and Technology Trends*, vol. 1, no. 2, pp. 56–70, 2020.