Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 171 (2020) 207–216

Procedia Computer Science

www.elsevier.com/locate/procedia

Third International Conference on Computing and Network Communications (CoCoNet'19)

NITCAD - Developing an object detection, classification and stereo vision dataset for autonomous navigation in Indian roads

Namburi GNVV Satya Sai Srinath, Athul Zac Joseph, S Umamaheswaran, Ch. Lakshmi Priyanka, Malavika Nair M, Praveen Sankaran

National Institute of Technology, Calicut, Kerala, India

Abstract

Autonomous vehicles with various levels of autonomy are becoming popular in developed countries due to their effectiveness in reducing the fatalities caused by road accidents. A developing country like India with the second largest population in the world, creates unique road scenarios for an autonomous car which requires a lot of testing and fine tuning before implementation. This leads to the importance of datasets providing information about various traffic situations in India. For planning its path ahead, autonomous vehicles have to detect, classify and estimate the depth of obstacles that they encounter on roads. The purpose of this paper is to provide a dataset for object classification, detection and stereo vision corresponding to Indian roads which can serve as a platform for developing effective algorithms for autonomous cars in Indian roads. In this work, we benchmarked the object classification by using confusion matrix obtained from various deep learning models, evaluated detection using Faster R-CNN and compared depth estimation processed by Realsense stereo camera by applying convolutional neural network based algorithms.

Keywords: Dataset, Object classification, Object detection, Stereo vision, Indian roads

1. Introduction

The future of transportation lies in the development of autonomous vehicles that can eliminate the factor of human error thereby reducing the chance of road accidents. The technology will only be effective if implemented across all the vehicles on the roads thereby reducing the uncertainties associated with drivers. For perfecting the technology to be implemented on a large scale, it should be tested across various traffic situations that will be encountered by these vehicles. Several large datasets like Imagenet [1] and COCO [2], are available for image classification but

* Corresponding author. Tel.: +0495-2286721.
E-mail address: psankaran@nitc.ac.in

10.1016/j.procs.2020.04.022

for localisation and segmentation, which are primary tasks for autonomous vehicles, only a few datasets like Pascal VOC [3] can be used. But it has a wide variety of classes that are not necessary for autonomous navigation. There are some datasets available exclusively for pedestrian detection such as Caltech Pedestrian Dataset [4], Citypersons dataset [5], Daimler Pedestrain [6]. Krause et al. [7] created a dataset which can be used for classification of cars. But autonomous vehicles need to detect different classes of obstacles present in its path. For that, Oxford robotcar dataset [8] has collected data in the city of Oxford, UK at various times in the same location. KITTI [9] is another dataset which is collected on the urban roads of Karlsruhe, Germany. Cityscapes [10], and Mapillary Vistas [11] created datasets for semantic understanding of urban streets. More recently Apolloscapes [12], BDD100k [13], nuScenes [14] provided datasets that are collected across various weather conditions, times and places, along with labels which are crowd sourced.

To test an autonomous vehicle in India, the above mentioned datasets cannot be used due to the lack of information about classes like auto rickshaws that are exclusively found on Indian roads. Also, unstructured traffic scenarios can be observed frequently on Indian roads. These factors make it essential to have a dataset exclusively for Indian roads which can give information regarding the same. Varma et al. [15] has studied about the situations in Indian roads and created a dataset named IDD. For an autonomous vehicle to plan its path, it should be aware of distances to other vehicles in its vicinity so that the velocity of other vehicles can be estimated. This can be achieved by using a 3D LiDAR which can scan its surroundings and give distances to different obstacles but this is rather expensive. A cost effective alternative is to use a stereo camera which can provide the depth information about the obstacles. The performance of this method purely depends upon the algorithms for evaluating depth information. Due to the introduction of better algorithms this method will be more suitable for a price conscious market like India.

In this context, there is a requirement for a new dataset, that can be used to develop autonomous navigation systems for Indian roads. So, a dataset named National Institute of Technology, Calicut Autonomous Driving (NITCAD) is presented in this paper. NITCAD primarily consists of NITCAD object dataset which can be used for classification, detection and NITCAD stereo vision dataset for depth estimation on Indian roads, thereby leading to the development of level 3 autonomous vehicles capable of handling Indian road scenarios.

2. Methodology

The autonomous vehicles that are being developed for Indian roads should be able to detect and classify different vehicle classes that are exclusively found here. This will be advantageous for performing the path planning operation since different vehicle classes behave differently on roads. NITCAD object dataset provided here was created under this objective.

Inorder to keep track of the detected objects on the road, the autonomous vehicle needs to evaluate the velocity of these objects. For developing stereo vision based velocity estimation algorithms, NITCAD stereo vision dataset provides image data collected from synchronised left and right cameras having global shutter.

The NITCAD object dataset was evaluated with different deep learning architectures and the respective confusion matrix, precision, recall values were found out. For the NITCAD stereo vision dataset, the relative difference between the disparity maps obtained by different methods are evaluated and interpreted.

2.1. Traffic scenario in India

India has one of the the highest number of road fatalities in the world. Autonomous navigation is one of the solutions to reduce the number of fatalities due to road accidents. To build an efficient and reliable system, the knowledge of the traffic structure in India is highly essential. Traffic in India is highly heterogeneous and the models trained for autonomous navigation in other countries may not be sufficient to characterize Indian traffic. Indias roads are functionally classified as expressways, national highways, state highways, district roads, and rural roads. Out of this, only expressways and some of the national highways are four-lane or six-lane. An example of unlaned traffic junction can be observed in Fig. 1(a). Most roads are unpaved with potholes and with ambiguous boundaries. Compared to other countries, India has low road density per 1000 people. This has led to many problems like traffic congestion and irregular traffic speeds. The Indian roads carry almost 90 per cent of the countrys passenger traffic. Passengers use a multitude of vehicles for transport daily which include cars, two-wheelers, and auto-rickshaws, etc. Auto-rickshaws

(a) An unlaned traffic junction.

(b) Unstructured environment prevalent in India.

Fig. 1: General traffic scenarios on Indian Roads.

are a class of vehicles that are truly unique to the Indian traffic system. Further, the frequency and variety of trucks and buses are also high as compared to other countries. Another huge bottleneck for autonomous navigation in India is that the pedestrians and drivers are less likely to follow traffic rules. Pedestrians often cross the road at arbitrary locations and drivers sometimes overtake from the wrong side. An example of this unstructuredness in traffic can be observed in Fig. 1(b).

2.2. Collection of data

To collect data, one RGB camera and a Stereo camera were mounted on a car and was made to travel in the rural and urban roads of Kerala where various traffic situations arise. The route in which the data was collected is shown in Fig. 2

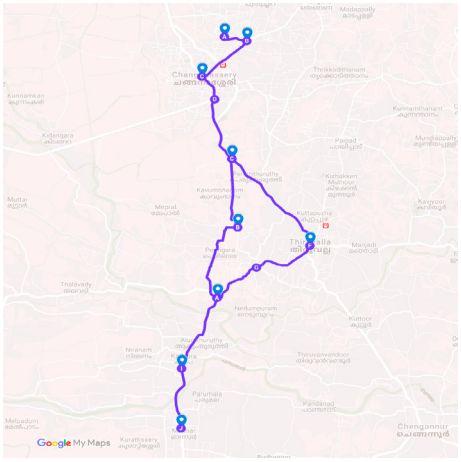
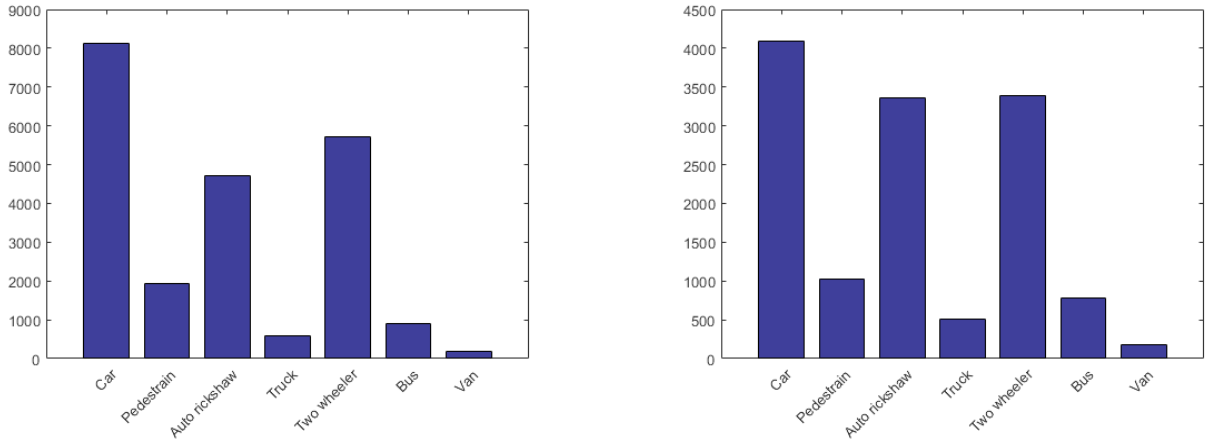


Fig. 2: Route followed for data collection.

2.3. NITCAD Object dataset

For training the system to classify different objects that could be encountered on an Indian road, a dataset including a variety of classes needs to be created. By using Noise Play 2 Action camera, traffic in and around Kottayam district, Kerala was recorded along the route shown in Fig.2 in 720p at 30fps. A set of images at a rate of 5 images per second were generated from this recorded video footage. These images were annotated using an online tool - Label box and a text file was created for each scene which has information about the location of various objects in that particular frame. These files can then be used for the visualization of the dataset as well as for training a system to detect and



(a) Number of objects per class.

(b) Number of images having a class.

Fig. 3: Details of NITCAD object dataset

Fig. 4: An example of various classes present in NITCAD object dataset. From left to right: Car, Bus, Pedestrian, Two wheeler, Truck, Van and Auto rickshaw

classify the objects on road. There are seven classes in the dataset namely car, pedestrian, auto rickshaw, truck, two wheeler, bus and van. A total of 11000 images were collected under different traffic conditions out of which 4800 images were manually labelled.

Fig. 3(a) gives details about the frequency of each class. From Fig. 3(a), it can be observed that the number of cars present in the dataset is maximum and the number of vans present is minimum. Fig. 3(b) gives details about the number of images or frames having a particular class from which it can be inferred that cars, auto rickshaws and two wheelers are almost present in all the frames while vans occur at rare instances on roads. Fig. 4 gives a typical example of all the classes present in our dataset.

2.3.1. Camera Calibration:

The images obtained from Noise play action camera were subjected to radial and tangential distortion because of the wide angle lens employed in the camera. For preserving the information in each frame the intrinsic camera

(a) distorted image.

(b) undistorted image.

Fig. 5: Camera Calibration results.

Table 1: Comparison with different datasets.

Dataset	Year	Avg.Resolution	Train/Validation/Split	Location	Unstructured
KITTI	2012	1245x375	7.5k/-/7.5k	Karlsruhe	
BDD100k	2017	1280x720	84M/36M	NY, SF	
Vistas	2017	1920x1080	18k/2k/5k	6 continents	
IDD	2019	1920x1080	31k/10k/4k	Hyderabad, Bangalore	✓
NITCAD	2019	1280x720*	4.8k/2.7k	Kerala	✓

* indicates the dimension before the undistortion operation. After undistortion, it is 1172x544.

matrix and distortion coefficients of the camera where computed according to Zhang [18] and were obtained as below:

$$\text{camera intrinsic matrix} = \begin{bmatrix} 791.6965 & 0 & 632.9851 \\ 0 & 791.4219 & 347.7182 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\text{radial distortion coefficients} = \begin{bmatrix} -0.3454 & 0.1593 & -0.0344 \end{bmatrix}$$

$$\text{tangential distortion coefficients} = \begin{bmatrix} 0.0021 & 0.0016 \end{bmatrix}$$

With the above obtained camera intrinsic matrix and distortion coefficients the images taken by Noise play action camera were undistorted and provided in the dataset. An example of distorted and corresponding undistorted image is shown in Fig.5 where the distortion is clearly visible near the edges. Around 10,000 undistorted images (of which 3600 are labelled) are also provided.

2.4. NITCAD Stereo vision dataset

For performing the task of depth estimation, data was collected in and around Kottayam district, Kerala using Intel RealSense Depth camera D435. This depth camera has 2 infrared cameras having global shutter that are triggered simultaneously so that calculation of disparity for a particular scene is possible. By using its inbuilt vision processor a disparity map can be generated which can be used for depth estimation. More efficient algorithms can be developed to improve its accuracy so that it becomes a cost effective depth approximation technique using stereo vision. Depth estimation obtained by the inbuilt vision processor can be used for validation of the results obtained after performing stereo algorithms. An example image is shown in Fig. 6 where a small disparity can be observed.

(a) Left image. (b) Right image.

Fig. 6: An example image pair from NITCAD stereo vision dataset.

Table 2: Evaluation of various architectures on NITCAD object dataset.

Architecture	Accuracy*	Precision	F1 score
DenseNet [19]	0.828	0.795	0.782
Inceptionv3 [20]	0.836	0.801	0.805
Mobilenet [21]	0.839	0.78	0.796
NASNet [22]	0.811	0.757	0.760
VGG16 [23]	0.789	0.779	0.750
Xception [24]	0.854	0.832	0.825

*Accuracy and Recall values are same as micro-averaging is considered for multiclass confusion matrix

3. Evaluation

NITCAD can be considered challenging only if the classes present in it are difficult to classify. Thus the dataset is processed accordingly and the cropped images are fed to various classification algorithms to obtain confusion matrix. To evaluate the detection algorithms on our dataset, Faster R-CNN is used. The stereo dataset is evaluated on the basis of average of the relative difference between the disparity maps generated (R_{diff}) by taking the output obtained from Intel RealSense as the ground truth.

3.1. NITCAD object dataset- Evaluation for classification

To evaluate how good the classification algorithms perform on the dataset, confusion matrix for different deep learning architectures was computed. Each image is cropped to extract all individual classes and a subset of the labelled data was considered for training, validation and testing. Pre-trained models on Imagenet were taken and initial layers were frozen as they learn about simple features like edges and lines and these are common in all the objects. Validation accuracy is used as a metric while training for 20 epochs and the best weights are used for testing. The confusion matrix for various architectures are represented in Fig. 7.

It can be inferred that the class van is being confused with car/auto by these architectures. Also, the auto-rickshaw which is the most common class in Indian roads has been classified well as the dataset collected has enough number of autos to train networks. The accuracy and precision values of various classes and models can be seen in Table 2 and Table 3

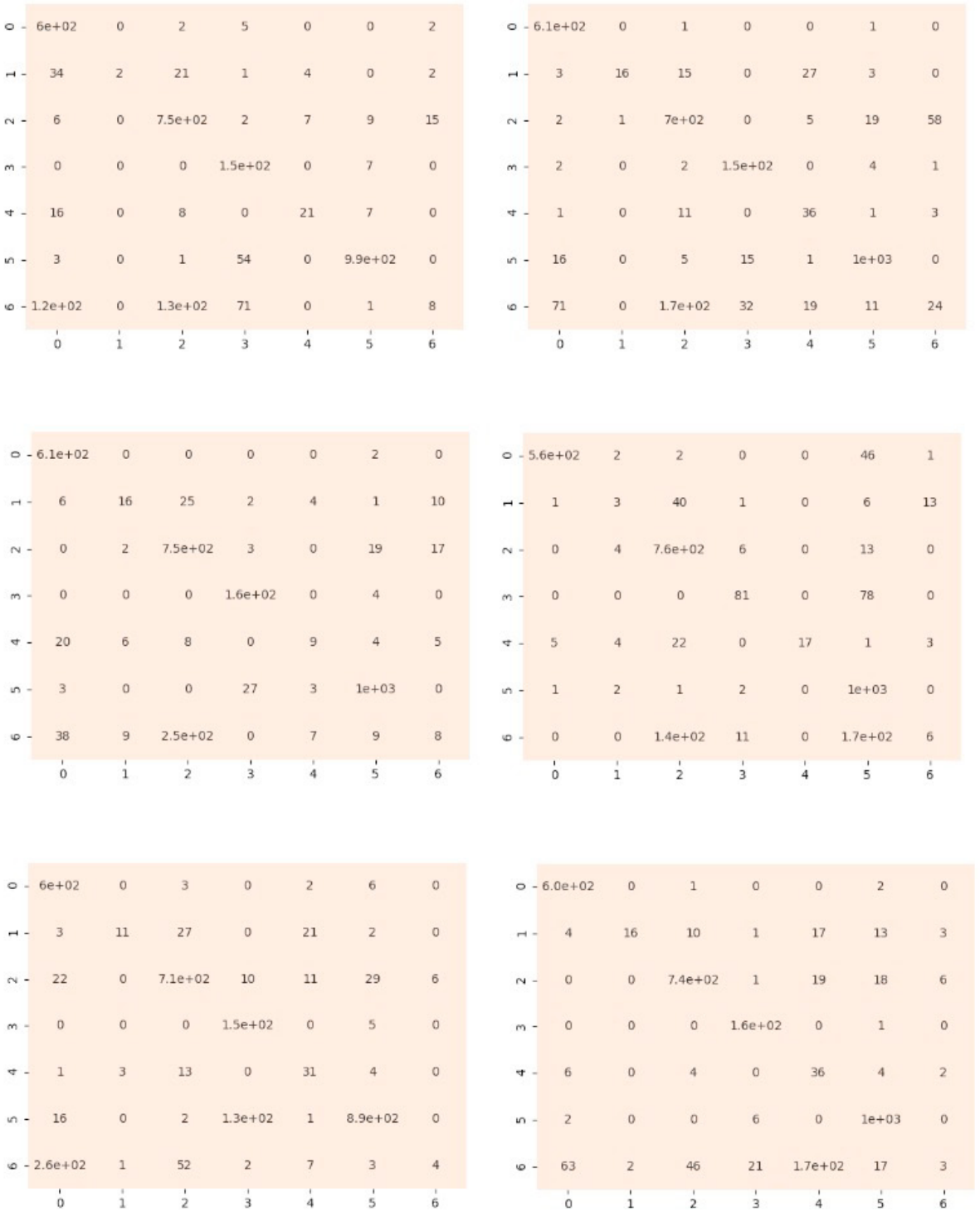


Fig. 7: Confusion matrix for various architectures.
DenseNet, Inceptionv3, Mobilenet, NASNet, VGG16 and Xception(from left to right)
0-Auto rickshaw, 1-Bus, 2-Car, 3-Pedestrian, 4-Truck, 5-Two Wheeler, 6-Van

Table 3: Precision, Recall values for different classes present in NITCAD object dataset.

Architecture	Precision							Recall						
	A	B	C	P	Tr	Tw	V	A	B	C	P	Tr	Tw	V
DenseNet	0.77	1.00	0.82	0.53	0.65	0.97	0.29	0.98	0.03	0.95	0.95	0.4	0.94	0.02
Inceptionv3	0.86	0.94	0.77	0.76	0.4	0.96	0.27	0.99	0.25	0.89	0.94	0.69	0.96	0.07
Mobilenet	0.9	0.48	0.72	0.82	0.39	0.96	0.2	0.99	0.25	0.94	0.97	0.17	0.96	0.02
NASNet	0.98	0.2	0.79	0.8	1.00	0.76	0.26	0.91	0.04	0.97	0.5	0.32	0.99	0.01
VGG16	0.66	0.73	0.87	0.51	0.42	0.94	0.4	0.98	0.17	0.9	0.96	0.59	0.85	0.01
Xception	0.88	0.88	0.92	0.84	0.14	0.94	0.21	0.99	0.25	0.94	0.99	0.69	0.99	0.009

A-auto rickshaw,B-bus,C-car,P-pedestrian,Tr-truck,Tw-two wheeler,V-van

3.2. NITCAD object dataset - Evaluation for detection

To evaluate the detection, Faster R-CNN is chosen. 1200 images are trained for 70 epochs with each epoch having 200 iterations in 4GB GPU system. Resnet is used as the base architecture to train, extract features and the metrics are tabulated in Table 4

Table 4: Different metrics obtained after training Faster R-CNN.

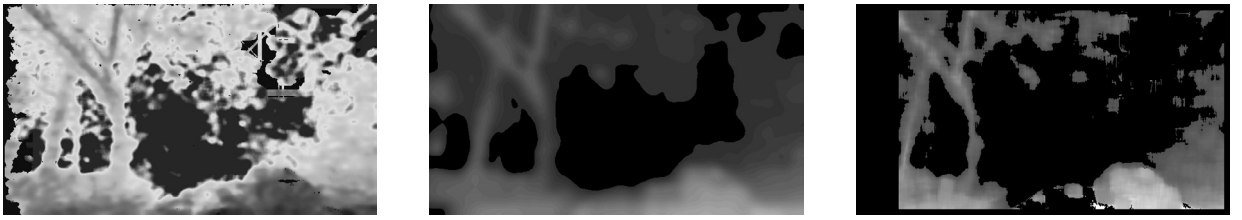
Classifier Accuracy	0.894
Loss RPN Classifier	0.057
Loss RPN Regression	0.0846
Loss Detector Classifier	0.26
Loss Detector Regression	0.11

3.3. Evaluation of NITCAD Stereo vision dataset

Intel Realsense stereo camera generates a depth map of the scene that is being recorded using the built in vision processor. This was taken as the ground truth of depth. For improving the depth estimation, a neural network based approach MC-CNN [16] was applied. For a pair of images corresponding to a scene MC-CNN generates a disparity map which is used to obtain the depth information. Pre-trained network on KITTI was chosen to estimate the disparity maps. Disparity map for that particular scene was also obtained using inbuilt functions provided by OpenCV library. Let $D_{Intel}(x, y)$ and $D_{method}(x, y)$ corresponds to the disparity maps generated by Intel Realsense stereo camera and by two of the above mentioned methods respectively for a particular scene. The average of the relative difference between the disparity maps generated(R_{diff}) can be obtained as

$$R_{diff} = \frac{1}{w \times h} \sum_{x \epsilon w, y \epsilon h} |D_{Intel}(x, y) - D_{method}(x, y)| \quad (1)$$

If the value of R_{diff} is less, then it can be implied that the disparity map obtained by the method is accurate. Obtaining the depth information is useful in estimating the velocity of the objects in the scene. The output obtained from Intel Realsense is taken as ground truth and the R_{diff} obtained with MC-CNN is 14 and with OpenCV is 18 where it is the average of about 100 image pairs.



(a) Intel Realsense (b) MC-CNN (c) OpenCV

Fig. 8: Disparity maps obtained from different algorithms of the image pair as shown in Fig.6. (Images thresholded for visual analysis)

4. Conclusion and Future Work

A challenging dataset is presented which includes various test cases that are frequently present in Indian roads. To get the information regarding velocity, a stereo dataset is presented which can be used to develop algorithms to obtain depth information. Various classes are labelled for object classification and is evaluated with confusion matrix which is obtained by different architectures. For detection, Faster R-CNN is used. The stereo dataset is evaluated by absolute sum of error between the output obtained from the Intel camera and with the methods described i.e MC-CNN and OpenCV. Our dataset can be further extended by collecting data which can include new classes like animals, lorry, sign boards etc. Research on the development of novel architectures that can detect and classify in various conditions including many edge cases needs to be carried on.

5. Acknowledgements

We would like to thank TEQIP - III for providing fund to acquire Intel RealSense D435 Depth camera.

References

[1] Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. pp. 248255. IEEE (2009)
[2] Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollr, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European conference on computer vision. pp. 740755. Springer (2014)
[3] Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. International journal of computer vision 88(2), 303338(2010)
[4] Dollar, P., Wojek, C., Schiele, B., Perona, P.: Pedestrian detection: A benchmark. In: Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. pp. 304311. IEEE (2009)
[5] Zhang, Shanshan, Rodrigo Benenson, and Bernt Schiele. "Citypersons: A diverse dataset for pedestrian detection." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017.
[6] Keller, Christoph Gustav, Markus Enzweiler, and Dariu M. Gavrila. "A new benchmark for stereo-based pedestrian detection." 2011 IEEE Intelligent Vehicles Symposium (IV). IEEE, 2011.
[7] Krause, Jonathan, et al. "3d object representations for fine-grained categorization." Proceedings of the IEEE International Conference on Computer Vision Workshops. 2013.
[8] Maddern, W., Pascoe, G., Linegar, C., Newman, P.: 1 year, 1000 km: The oxford robotcar dataset. IJ Robotics Res. 36(1), 315 (2017)
[9] Geiger, Andreas, et al. "Vision meets robotics: The KITTI dataset." The International Journal of Robotics Research 32.11 (2013): 1231-1237.
[10] Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U.,Roth, S., Schiele, B.: The cityscapes dataset for semantic urban scene understanding. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3213 3223 (2016)
[11] Neuhold, G., Ollmann, T., Bul, S.R., Kontschieder, P.: The mapillary vistas dataset for semantic understanding of street scenes. In: International Conference on Computer Vision (ICCV) (2017)
[12] Huang, Xinyu, et al. "The apolloscape dataset for autonomous driving." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. 2018.
[13] Yu, Fisher, et al. "BDD100K: A diverse driving video database with scalable annotation tooling." arXiv preprint arXiv:1805.04687 (2018).
[14] Caesar, Holger, et al. "nuScenes: A multimodal dataset for autonomous driving." arXiv preprint arXiv:1903.11027 (2019).
[15] Varma, Girish, et al. "IDD: A dataset for exploring problems of autonomous navigation in unconstrained environments." 2019 IEEE Winter Conference on Applications of Computer Vision (WACV). IEEE, 2019.

[16] J. Zbontar, Y. LeCunet al., "Stereo matching by training a convolutional neuralnetwork to compare image patches.Journal of Machine Learning Research,vol. 17, no. 1-32, p. 2, 2016.
[17] E. Rosten and T. Drummond, "Machine learning for high-speed cornerdetection, in European Conference on Computer Vision, vol. 1, May 2006,pp. 430443. [Online]. Available: http://www.edwardrosten.com/work/rosten2006machine.pdf
[18] Z. Zhang, "A flexible new technique for camera calibration,IEEE Transactions onpattern analysis and machine intelligence, vol. 22, 2000
[19] Huang, Gao, et al. "Densely connected convolutional networks." Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.
[20] Szegedy, Christian, et al. "Rethinking the inception architecture for computer vision." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.
[21] Howard, Andrew G., et al. "Mobilenets: Efficient convolutional neural networks for mobile vision applications." arXiv preprint arXiv:1704.04861 (2017).
[22] Zoph, Barret, et al. "Learning transferable architectures for scalable image recognition." Proceedings of the IEEE conference on computer vision and pattern recognition. 2018.
[23] Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." arXiv preprint arXiv:1409.1556 (2014).
[24] Chollet, Franois. "Xception: Deep learning with depthwise separable convolutions." Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.