# Evaluation of 3D Object Detection Methods in Autonomous Driving using the KITTI Dataset

Anthony L. Shen[1]*, Chien E. Lin[2]

*[1]Northgate High School, Walnut Creek, California, United States, 94598*
*[2]Department of Robotics, University of Michigan, Ann Arbor, Michigan, United States, 48109*

KEYWORDS. Evaluation, Autonomous Driving, 3D Object Detection Methods

BRIEF. This study provides comparison and evaluation of three 3D object detection methods and finds the one that achieves the best precision and accuracy.

*Abstract.* A number of research papers have been published on 3D Object Detection methods of autonomous driving. However, these papers often describe approaches to specific problems or scenarios without providing comparison and evaluation of the existing methods and their limitations. The aim of the study is to compare and evaluate three methods for the 3D object detection benchmark in autonomous driving: Deep Learning and Geometry [1], Triangulation Learning Network [2], and Monocular 3D Object Detection [3]. Each of the three methods processes the left color images and camera calibration matrices from the KITTI dataset, and the results are compared with the training labels. By using the KITTI metric [4] and analyzing each of the method's program codes, the performance, accuracy, and techniques are evaluated. The results demonstrate the reliability and accuracy of the Triangulation Learning Network [2] for 3D object detection, and it outperforms the other two methods. Finally, the article provides a general discussion of the three novel approaches for 3D object detection and insight into the development of better concepts and methods for this benchmark.

## INTRODUCTION.

Autonomous vehicles and reliable self-driving machines have become more and more relevant in our everyday society. They decrease traffic jams, reduce accidents, and result in less fuel being wasted. Researchers have already begun implementing and improving self–driving cars and models. They use benchmarks such as optical flow, visual odometry, 3D object detection, and 3D tracking to evaluate the accuracy of their model and have made vast improvements in the world of autonomous driving.

Out of all these benchmarks, 3D object detection is the most important and indispensable part of the perception system. 3D object detection is a computer vision task that involves identifying and localizing objects (finding their orientation and position) in 3D space from sensory inputs, such as images, LiDAR data, or both. Through 3D object detection, autonomous vehicles can make real-time decisions to avoid collisions and ensure safe driving behavior, putting less stress and restrictions on the driver. Furthermore, with the introduction of deep learning, the task of object detection has grown significantly in terms of speed and accuracy, making it the most important and fascinating benchmark to research and evaluate in self-driving models.

This paper seeks to investigate three different approaches to the 3D Object Detection benchwork including: Deep Learning and Geometry [1], Triangulation Learning Network [2], and Monocular 3D Object Detection [3]. It will discuss the benefits and drawbacks of the methods based on the techniques used in each program and the results from testing on the KITTI dataset [5], and provide an overall evaluation of each method.

## MATERIALS & METHODS.

To begin testing and evaluating the approaches for 3D object detection, there must be a general understanding for each method.

First, Deep Learning and Geometry [1] presents an approach for 3D object detection and pose estimation from a single image. It obtains relatively accurate 3D object properties using a deep neural network and then combines the approximations with constraints provided by a 2D bounding box to form a 3D bounding box. The first network output approximates the 3D object orientation, and the second network obtains the 3D object dimensions. These estimates, integrated with the constraints provided by the 2D bounding box, can recover a solid and accurate 3D pose. This method is simpler compared to the other two methods because it does not require preprocessing stages or 3D object models.

The next method, Triangulation Learning Network [2], effectively utilizes stereo information resulting in lower costs for hardware and can adapt to different scales of objects. It employs 3D anchors to establish correspondences between the regions of interest in stereo images, from which the neural network learns to detect and triangulate the targeted object in 3D space. Additionally, it has a cost-effective channel reweighting strategy that biases the network towards key parts of the object and benefits triangulation learning.

Finally, the monocular approach [3] performs 3D object detection from a single monocular image. The method seeks to generate a set of candidate class-specific object proposals, which are run through a convolutional neural network to obtain high-quality detections. In particular, it places object candidates in 3D, and then scores each candidate box displayed to the image plane via several intuitive potentials. They are then further processed by a convolutional neural network resulting in a fast 3D object detection.

Overall, these methods each provide unique approaches to object detection and are beneficial in their own way. Next, are the steps for preparing the data, obtaining and tweaking the autonomous driving programs, and running the code on Google Colab.

### *Dataset and Data Processing.*

The KITTI dataset was used to test the three methods. The KITTI dataset is a widely used benchmark for autonomous driving tasks, such as stereo vision, optical flow, scene flow, visual odometry, and 3D object detection. The dataset consists of high-resolution images and videos captured by a calibrated camera mounted on a car. It covers diverse urban scenarios and driving conditions, such as highways, residential areas, city centers, and country roads. The KITTI does not cover all environments such as rural and aerial views, however it overall provides a realistic and challenging testbed for evaluating and comparing different methods for autonomous driving applications.

We used the 3D Object Detection 2017 data including the left color images, rights color images, camera calibration matrices, and the training labels of the object data set. The methods would generate results from processing the left and right color images and camera calibration matrices. The training labels, which contained the correct object name and exact location, dimensions, and orientation of the 3D bounding box, would be used to compare the results of each method. Because there

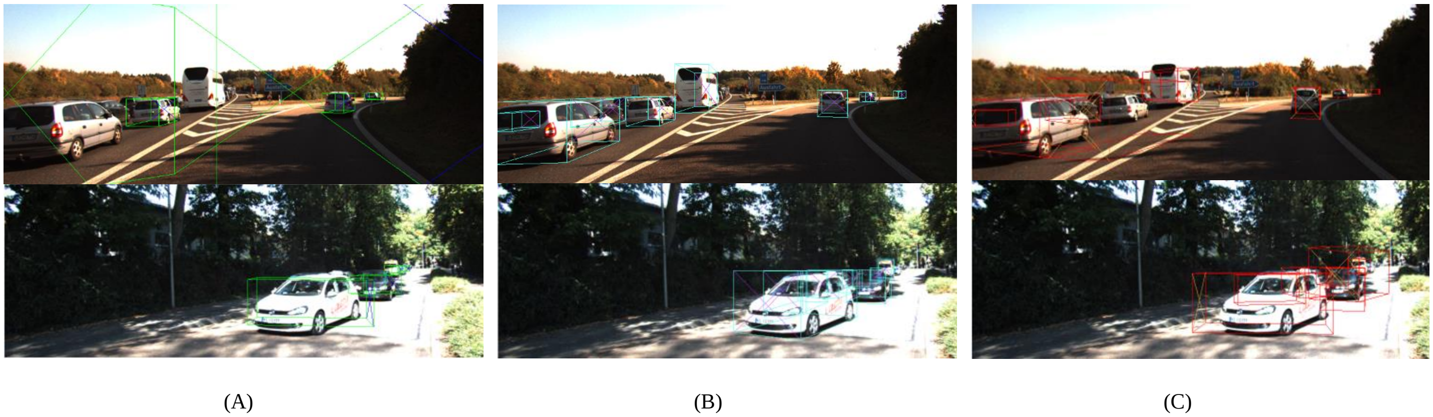

**Figure 1:** Comparison between the three methods for object detection in two testing images – the 007475.png (top three panels) and the 007479.png (bottom three panels). (A) Deep Learning & Geometry: 4 car objects detected in top image (007475.png) and 4 car objects detected in bottom image (007479.png). (B) Triangulation Learning Network: 7 car objects & 1 bus object detected (top) and 6 car objects & 1 bus object detected (bottom). Monocular 3D Object Detection: 7 car objects & 1 bus object detected (top) and 6 car objects & 1 bus object detected (bottom).

were over 7400 training images, camera calibration files, and training label files, we decided to take a subset of the data. we used 100 pairs of training images and camera calibration files, as well as 100 of the corresponding training labels to test and compare the results.

*AI Model.*

Open-source programs for the three autonomous driving models were obtained from GitHub and imported into Google Colab. Changes in the programs were made to match the environment of Google Colab and to provide the key data and results.

First, efforts were made to optimize the hyperparameters, however there was little to no change in performance for each of the three methods indicating that optimization was reached. The three programs already had the most optimal hyperparameter settings targeting the accuracy, speed, and memory usage of each method. The hyperparameters involved in these 3D object detection methods included:

1) The learning rate, batch size, weight decay, and optimizer for training the model.

2) The type and parameters of the region proposal network, such as anchor shapes, scales, and ratios.

Next, in each method, there was a main program that utilized modules including torch_lib.Dataset, library.Math, library.Plotting, and torch_lib.ClassAverages. The modules were added as program files below each of the main programs to make them executable in Google Colab. Next, data files were imported to each of the three Google Colabs including a camera calibration matrix file (with 100 data informations of traffic) for testing, pretrained weight files for reliable functioning and results, and a label dataset file to evaluate the resultsThen, each of the programs were truncated, and unnecessary variables and code segments were deleted. Functions were tweaked to provide key data about the 3D bounding box including: the object detected (i.e. pedestrian), the azimuth value, the coordinates of the bounding box, and its dimensions. In a separate program, the calculated results from the methods were stored and compared with the labels. Finally, methods were scored based on the KITTI metric. The official KITTI metric [4] for the evaluation calculated the following: Average Orientation Estimation (AOS), Average Precision (AP), and Orientation Score (OS). The Average Orientation Estimation is a value between 0 and 1, where 1 represents a perfect prediction. The Average Precision is a value between 0 and 1 that evaluates the localization algorithm and the performance of the object detection, and is calculated under the area of the precision recall curve. The Orientation score is the ratio of AOS over AP and represents the error averaged across all test images.

RESULTS.

According to Table 1, which summarizes the results for the three methods based on the KITTI metric [4], the Deep Learning and Geometry [1] method had the least AOS, Average Precision, and Orientation Score with values of 0.8523, 0.8678, and 0.9821 respectively. The Monocular 3D Object Detection [3] performed decently with values of 0.9165, 0.9204, and 0.9958 for each of the three metrics. Finally, the Triangulation Learning Network [2] had the highest percentages for AOS and AP, 0.9434 and 0.9467, and the highest Orientation Score, 0.9965. Figure 1 shows the number and type of objects detected and the orientation and position of the predicted bounding boxes for the three approaches.

DISCUSSION.

Overall, the Deep Learning and Geometry [1] method detected the fewest objects and had the least precision and accuracy compared to the other methods. Its AOS, Average Precision, and Orientation Score were the lowest. The Monocular 3D Object Detection [3] performed better, with around a 6% increase in both AOS and Average Precision. The Triangulation Learning Network ultimately performed the best with the highest AOS, Average Precision, and Orientation Score. This method was able to consistently and accurately identify the position and type of most if not all objects: cars, buses, pedestrians, etc.

Other articles and studies demonstrate the same results, and the Triangulation Learning Network [2], overall, is shown to be the best approach so far. However, the three methods each have novel and interesting approaches to identify objects and create 3D bounding boxes.

**Table 1**. 3D Object Detection Evaluation for Three Autonomous Driving Methods

| Method | AOS | AP | OS |
|---|---|---|---|
| Deep Learning & Geometry [1] | 0.8523 | 0.8678 | 0.9821 |
| Triangulation Learning Network [2] | 0.9434 | 0.9467 | 0.9965 |
| Monocular 3D Object Detection [3] | 0.9165 | 0.9204 | 0.9958 |

Deep Learning and Geometry [1] can recover relatively accurate 3D bounding boxes for known object categories from a single view. Using a MultiBin loss for orientation prediction and an effective choice of box dimensions as regression parameters, the method estimates stable and accurately-positioned 3D bounding boxes without additional 3D shape models or sampling strategies with difficult pre-processing pipelines.

For the Triangulation Learning Network [2], the 3D bounding boxes predicted by the baseline network and the stereo method are presented in the above image in Figure 1B. In general, the predicted blue bounding boxes match the ground truths and labels very well when the Triangulation Learning Net is integrated into the base-line model. The method reduces depth error, especially when the targets are far away from the camera. Object targets missed by the baseline are successfully detected. For example, some subtle cars in the middle of the top image are detected as well as some heavily truncated cars in the right of the bottom image, since the object proposals are in 3D, regardless of 2D truncation. Overall, the Triangulation Learning Network [2] presents a novel network for performing accurate 3D object detection using stereo information. It includes a solid baseline monocular detector, which is extended to stereo by combining with the proposed Triangulation Learning Net. The network learns to triangulate the targeted object in a forward pass. It also introduces an efficient channel reweighting method to emphasize informative features and weaken unnecessary signals. All of this form the base-line detector and achieve great performance.

There were a few limitations that may have caused some form of error in the results of the three methods. Namely, we used a small subset of the 7400 test images, camera calibration matrices, and training labels, 100 of each (because of limited run time for results). In addition, there wasn't a huge amount of training done before methods were run and the weights were only close estimates. In our next steps towards autonomous research, we will analyze more 3D Object Detection methods and use more datasets not just from KITTI, but also from other sites, such as the PASCAL VOC dataset. This will provide a further understanding of the methods used in autonomous driving and help with obtaining more accurate results and evaluations of their performance.

## CONCLUSION.

In conclusion, the Triangulation Learning Network [2] was the best out of the three methods. Through specific techniques and algorithms, it identified more objects with better precision and accuracy. In general, the research provided a thorough understanding of the three methods and the essential concepts and ideas of each. Further research and evaluation of autonomous methods will lead to the development of better perception systems in autonomous driving and allow cars to handle more diverse scenarios, environments, and system configurations.

## ACKNOWLEDGEMENTS.

I would like to thank the Veritas AI Fellowship program and my mentor, Chien E. Lin, for guiding me through this project, and helping with AI research, programming, and the final manuscript.

## REFERENCES.

1. A. Mousavian, D. Anguelov, J. Flynn, 3d Bounding Box Estimation Using Deep Learning and Geometry, *2017 IEEE Conference on Computer Vision and Pattern Recognition*, 7040 - 7079 (2017).
2. Z. Qin, J. Wang, Y. Lu, Triangulation Learning Network: from Monocular to Stereo 3D Object Detection, *2019 IEEE Conference on Computer Vision and Pattern Recognition*, 1 – 8 (2019).
3. X. Chen, K. Kundu, Z. Zhang, Monocular 3D Object Detection for Autonomous Driving, *2016 IEEE Conference on Computer Vision and Pattern Recognition*, 2147 - 2154 (2016).
4. A. Geiger, P. Lenz, R. Urtasun, Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite, *2012 IEEE Conference on Computer Vision and Pattern Recognition*, 3357 – 3359 (2012).
5. A. Geiger, P. Lenz, R. Urtasun, *3D Object Detection Evaluation 2017*, *The KITTI Vision Benchmark Suite* (2012); https://www.cvlibs.net/datasets/kitti/eval_object.php?obj_benchmark=3d.

Anthony Shen is a student at Northgate High School in Walnut Creek, CA; he participated in the Veritas AI Fellowship program.