NAVARCH 565 FA 2023 Final Project

Yanxi Lin
shyyl@umich.edu

Qi Dai
qidai@umich.edu

Hanxi Wan
wanhanxi@umich.edu

***Abstract*—3D object detection has emerged as a pivotal challenge in the realm of computer vision, particularly in applications like autonomous driving and robotics. The project delves into MonoCon, a state-of-the-art neural network framework designed to tackle this challenge using a single camera input. This method proposes a simple but effective formulation for monocular 3D object detection without exploiting any extra information such as lidar and depth. It learns Monocular Contexts, as auxiliary tasks in training, to help monocular 3D object detection. We revised and tested MonoCon in the KITTI benchmark on the car category under different scenarios and output 3D bounding box prediction to evaluate the performance.**

I. Introduction

The field of computer vision has witnessed significant advancements in recent years, with a particular focus on 3D object detection, which is a crucial component in many advanced systems, such as autonomous vehicles and robot navigation. Traditionally, this task relies on depth-sensing technologies like LiDAR or stereo cameras. However, these methods come with limitations in terms of cost and complexity. Monocular 3D object detection offers a more cost-effective and simplified alternative, using only a single camera to perceive depth and dimensions of objects. This simplicity makes it particularly attractive for widespread applications.

Recognizing the limitations of multi-sensory approaches, the research focus shifted to monocular image-based techniques. This shift was driven by the need for simpler, more cost-effective solutions without compromising detection accuracy. Techniques like M3D-RPN [1] and MonoDIS [4] paved the way, utilizing advanced algorithms to extract depth information from single images. However, these methods still grappled with challenges in accuracy, particularly in complex traffic scenarios.

Early monocular approaches relied heavily on handcrafted features and geometric constraints. However, with the advent of deep learning, there has been a paradigm shift. Convolutional Neural Networks (CNNs) and later, advanced architectures like R-CNNs, have been employed for feature extraction and object localization in 2D space, with subsequent steps to estimate depth and 3D dimensions.

MonoCon marked a significant advancement in this domain. It presents a method for monocular 3D object detection without extra informationS. MonoCon focuses on learning monocular contexts as auxiliary tasks during training, enhancing performance in 3D object detection. It uses a Deep Neural Network (DNN) based feature backbone, several regression head branches for 3D bounding box prediction, and branches for learning auxiliary contexts, which are discarded after training for efficiency. This method not only improves the precision in estimating the dimensions and orientations of vehicles but also does so with remarkable speed and efficiency. Comparisons with earlier methods demonstrate MonoCon's superior performance, especially in scenarios with diverse vehicle types and orientations.

Despite its advancements, MonoCon faces challenges, particularly in handling scenarios with partial occlusions, varying lighting conditions, and accurately detecting distant objects. These limitations highlight the ongoing need for refinement in monocular 3D detection methods, especially in diverse and unpredictable real-world environments.

The field of monocular 3D detection is rapidly evolving, with ongoing research focused on addressing the limitations of current methodologies. Future enhancements are expected to leverage emerging technologies in AI and deep learning, potentially leading to more robust and adaptable 3D detection systems. The integration of these technologies could significantly improve detection accuracy in varied operational conditions, marking a new era in computer vision and autonomous vehicle technology.

This report aims to provide a comprehensive overview of MonoCon. It will explore the technical foundations of the approach, including its architecture and the integration of geometric principles. This paper will also compare our revised MonoCon's performance with the original version, explaining the improvements and highlighting its strengths and limitations.

II. Approach

A. MonoCon Method

The overall structure of the original MonoCon method proposed by Liu et al [2] is illustrated in Fig.1, which consists of three main parts: feature backbone, 3D bounding boxes prediction and auxiliary contexts generation. Given a RGB image input, the feature backbone produces a feature map. Then, two light-weight regression head branches are used for 3D bounding boxes prediction and auxiliary contexts generation respectively.

1) Feature Backbone: We use the DLA-34 network [5] as the feature backbone, which is widely used in monocular 3D object detection. Given a RGB image input with $3 \times H \times W$ dimensions, the feature backbone produces a feature map with $D \times h \times w$ dimensions, where D is the feature map dimension, $h = H/s$ and $w = W/s$, with s being the feature backbone's stride/sub-sampling.

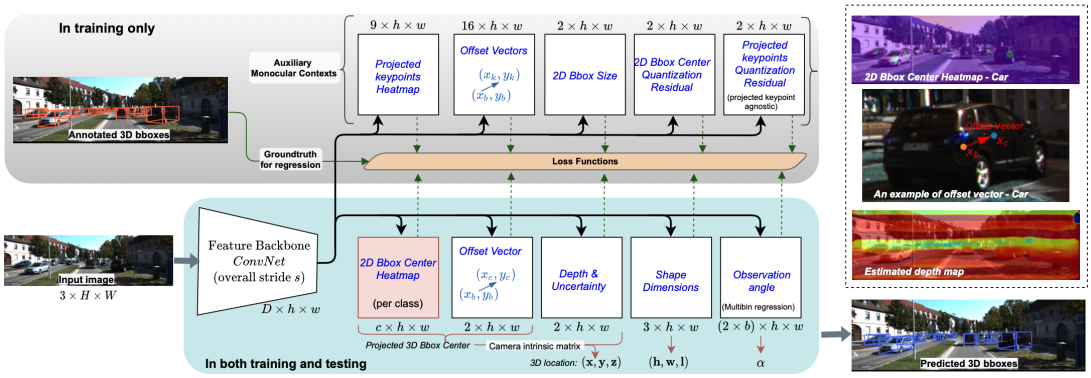

Fig. 1: Illustration of the original MonoCon method for monocular 3D object detection without additional information.

2) 3D Bounding Box Prediction: We use a regression head to compute the 2D bounding box center heatmap H^b with dimensions $c \times h \times w$ per class, where c is the number of classes. The center for the 2D bounding box (x_b, y_b) for each class is then determined based on the peaks in each channel of H^b after Non-Maximum Suppression (NMS) and thresholding. Using the light-weight regression head proposed by Liu et al [2], the offset vector from (x_b, y_b) to the center of 3D bounding box (x_c, y_c), the corresponding depth z, the shape dimensions (h, w, l) and the observation angle α are retrieved.

3) Auxiliary Contexts: We consider four types of projection information from 3D bounding box as auxiliary contexts: (i) the heatmaps of the 9 projected keypoints (8 corner points and 1 center point), (ii) the offset vectors for the 8 projected corner points from (x_b, y_b), (iii) the 2D bounding box size, and (iv) the quantization residual of a keypoint location.

B. Loss Functions

The overall loss function we use is a combination of five widely used loss functions in monocular 3D object detection with trade-off weights, which consists of (i) the Gaussian kernel weighted focal loss for heatmaps, (ii) the Laplacian aleatoric uncertainty loss function for the depth, (iii) the dimension-aware L1 loss function for shape dimensions, (iv) the standard cross-entropy loss function for the bin index in observation angles, and (v) the standard L1 loss function for offset vectors, the intra-bin angle residual in observation angles, 2D bounding box sizes, and the quantization residual.

III. Experiments

In this section, we tested the MonoCon in the widely used and challenging KITTI 3D object detection benchmark.

A. Data

The KITTI dataset consists of 1,229 images for training and 760 images for testing. The only category of interest is car. It is composed of a training set, a validation set, and a testing set, where the validation set and training set are split from the training images and they both have ground truth labels. The training set is used for training and the performance on the validation set is an indicator of performance for development.

Besides the commonly used data augmentation methods including photo-metric distortion and random horizontal flipping introduced in the original MonoCon [2], we also applied effects that aimed at simulating different weather [3]. These effects include adding fog and adding sun flare. These effects help the model generalize in different weather conditions given the limited training dataset.

B. Evaluation Metrics

Following the evaluation protocol of the official KITTI 3D object detection, we used the average precision (AP) to evaluate the 3D bounding boxes prediction.

$$AP_{3D|R40}@IoU = 0.5(moderate) \tag{1}$$

The AP of 3D bounding boxes is computed with the intersection-over-union (IoU) threshold of 0.5 with 40 recall positions under moderate difficulty settings for the car.

C. Implementation Details

We trained MonoCon on a single NVIDIA GeForce RTX 4090 GPU with a batch size of 8 for 200 epochs. The initial learning rate is 2.5e-4 and the weight decay is 1e-6.

D. Results

1) Prediction Visualization: We visualize the prediction of our model in two scenarios different in weather. The first one features a sunny day, while the second one is a foggy day.

Figures 2, 3, 4 show the 2D Bounding Box, 3D Bounding Box, and Bird Eye's View prediction results. It can be seen that our model successfully detected all the cars in this scenario.

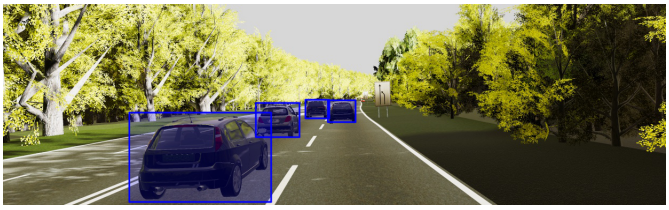

Fig. 2: Sunny - 2D Bounding Box Prediction

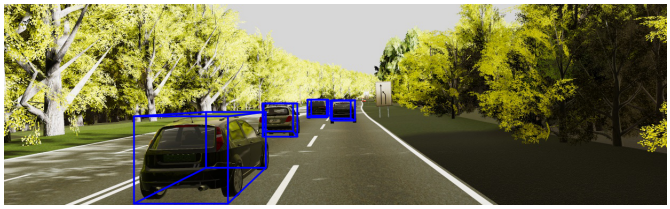

Fig. 3: Sunny - 3D Bounding Box Prediction

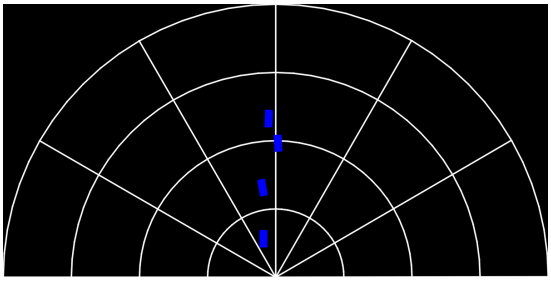

Fig. 4: Sunny - Bird Eye's View Prediction

Figure 5, 6, 7 show the 2D Bounding Box, 3D Bounding Box, and Bird Eye's View prediction results in foggy weather. Our model has demonstrated the ability to detect most of the cars in this scenario, which is a promising outcome. However, it appears to be struggling with recognizing a vehicle at a distance, which is understandable given that it is small and blurry even for human eyes. Despite this limitation, the overall performance of our model in this foggy scenario is acceptable and shows potential for further improvement.

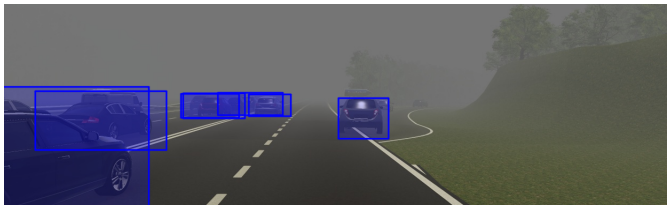

Fig. 5: Foggy - 2D Bounding Box Prediction

Fig. 6: Foggy - 3D Bounding Box Prediction

The original Monocon method reaches an average precision of 6.8316 in the foggy scenario. Our enhanced model reaches an average precision of 12.5778 in the foggy scenario.

2) Hyper-parameters Comparison: We also compared the performance under different parameter values, noting that we only changed one parameter per time.

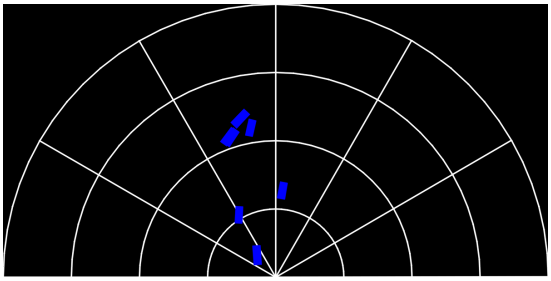

Fig. 7: Foggy - Bird Eye's View Prediction

Learning rate	$Car, AP_{3D\|R40}@IoU \geq 0.7$		
	Easy	Moderate	Hard
2.25×10^{-4} (Original)	**54.55**	**54.05**	**51.68**
2.25×10^{-5}	3.48	2.64	2.40
2.25×10^{-6}	55.24	48.28	46.26

TABLE I: Car detection accuracy under different learning rate.

Weight decay	$Car, AP_{3D\|R40}@IoU \geq 0.7$		
	Easy	Moderate	Hard
1×10^{-4}	**84.68**	**84.31**	**79.40**
1×10^{-5} (Original)	54.55	54.05	51.68
1×10^{-6}	69.96	70.31	68.08

TABLE II: Car detection accuracy under different weight decay.

Backbone Layer Number	$Car, AP_{3D\|R40}@IoU \geq 0.7$		
	Easy	Moderate	Hard
34 (Original)	54.55	54.05	51.68
46	61.83	62.91	59.45
60	**75.00**	**73.08**	**69.07**

TABLE III: Car detection accuracy under different backbone layer number.

The original parameter configuration of the MonoCon model yields an average precision of 19.7058, as calculated using Eq. 1. By optimizing the parameters based on the best-performing values found in Tables I, II, and III, the average precision is enhanced to 20.0038.

IV. Conclusion

In this report, we introduced and delved into MonoCon, a method for monocular 3D object detection. MonoCon leverages monocular contexts, such as projected keypoints, as auxiliary tasks to improve the performance of bounding box prediction.

We enhanced the model's ability to generalize in different weather conditions using data augmentation. We also analyzed the influence of different hyperparameters. By visualizing the prediction and comparing the average precision, we demonstrated that our model achieves competitive accuracy and efficiency in various scenarios.

However, we also acknowledge the limitations of MonoCon, such as handling varying lighting and distant objects. We suggest some possible directions for future research, such as incorporating attention mechanisms and temporal information to enhance the monocular 3D detection system.

References

[1] G. Brazil and X. Liu. M3d-rpn: Monocular 3d region proposal network for object detection, 2019.

[2] X. Liu, N. Xue, and T. Wu. Learning auxiliary monocular contexts helps monocular 3d object detection, 2021.

[3] U. Saxena. Automold road augmentation library. https://github.com/UjjwalSaxena/Automold--Road-Augmentation-Libra, 2022.

[4] A. Simonelli, S. R. R. Bulò, L. Porzi, M. López-Antequera, and P. Kontschieder. Disentangling monocular 3d object detection, 2019.

[5] F. Yu, D. Wang, E. Shelhamer, and T. Darrell. Deep layer aggregation, 2019.