Star-Convolution for Image-Based 3D Object Detection

Yuxuan Liu, Zhenhua Xu and Ming Liu

Abstract— **3D object detection with only image inputs is an interesting and important problem in computer vision and autonomous driving. Nowadays, most existing monocular 3D object detection algorithms rely solely on the approximation power of convolutional neural networks to learn a mapping from pixels to 3D predictions without knowing the projection matrix of the camera. To endow the networks with camera projection knowledge, we propose the Star-Convolution module for application to image-based 3D detection. The introduced module increases the receptive field of the detector and embeds the camera's projection geometry inside the network while keeping the network end-to-end trainable. We test the module with different baselines in both monocular and stereo 3D object detection, and we achieve significant improvements on both tasks. The code will be published in https://github.com/Owen-Liuyuxuan/visualDet3D.**

I. Introduction

Detecting objects of interest and estimating their 3D locations, orientations, and sizes with only images is an important problem in computer vision and autonomous driving. Compared to LiDAR, visual sensors such as cameras are cheaper, smaller, consume less energy, and are much easier to integrate into the mechatronic system of a robot. As a result, even though cameras do not provide accurate distance measurements of the environment like LiDARs do, 3D detection with monocular or stereo cameras has been drawing increasing attention among researchers.

As a corollary to this lack of absolute scale measurements in the input, performing 3D object detection with only one image frame is an ill-posed problem. Many recent works have directly applied convolutional neural networks (CNNs) to learn a mapping from image features to the depth of the objects [1], [2], [3]. Under such formalism, the 3D distances between the objects and the camera are computed from a sequence of local operations such as convolution on the input image pixels. We note that the calibration matrix or projection geometry is absent under such a CNN-based paradigm. These works depend solely on the expressivity of the neural network to "learn" a functional mapping between image pixel inputs and the distance measurements.

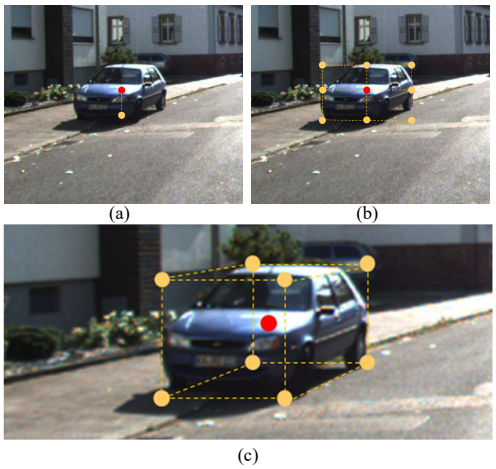

Fig. 1: The sampling point of (a) the GAC module; (b) the 2D Star-Convolution module; (c) the proposed 3D Star-Convolution module. The proposed 3D Star-Convolution extracts more sophisticated features from the initially predicted keypoints.

Brazil *et al.* [3] proposed to utilize the statistical prior in the anchors to normalize the depth prediction targets. Such a strategy improves the convergence speed and significantly boosts the prediction accuracy for close-up objects. Taking a closer look into the anchors' priors, we find that the 3D priors of the anchors are implicitly determined by the projective geometry of the camera and the dimensions of the target objects [4]. The performance improvements achieved by [3] and [4] demonstrate how additional geometric priors boost the performance of monocular 3D detectors, even when only applied in the final decoding phase.

However, most existing monocular 3D detectors perform learnable inference solely in the image-view, with mostly position-invariant operators. Thus, explicitly integrating the geometric knowledge into the inferencing and even the training phase of the convolutional neural network has long been the intuitions behind significant advancements in monocular 3D object detection.

SS3D [5] apply non-linear optimization online to minimize the reprojection error between the network's 2D and 3D prediction, and supervise the result of the non-linear optimization with ground truth, which includes the camera parameters in the backpropagation step. However, the inference pipeline of the learnable

*This work was supported by the National Natural Science Foundation of China (Grant No. U1713211), the Research Grant Council of Hong Kong SAR Government, China, under Project No. 11210017, and No. 21202816, and Shenzhen Science, Technology and Innovation Comission (SZSTI) JCYJ20160428154842603, awarded to Prof. Ming Liu.

The authors are with the Robotics and Multi-Perception Laborotary, Department of Electronic and Computer Engineering, The Hong Kong University of Science and Technology. email:yliuhb@connect.ust.hk, zxubg@ust.hk, eelium@ust.hk.

Manuscript 628 submitted to 2022 IEEE International Conference on Robotics and Automation (ICRA). Received September 14, 2021.

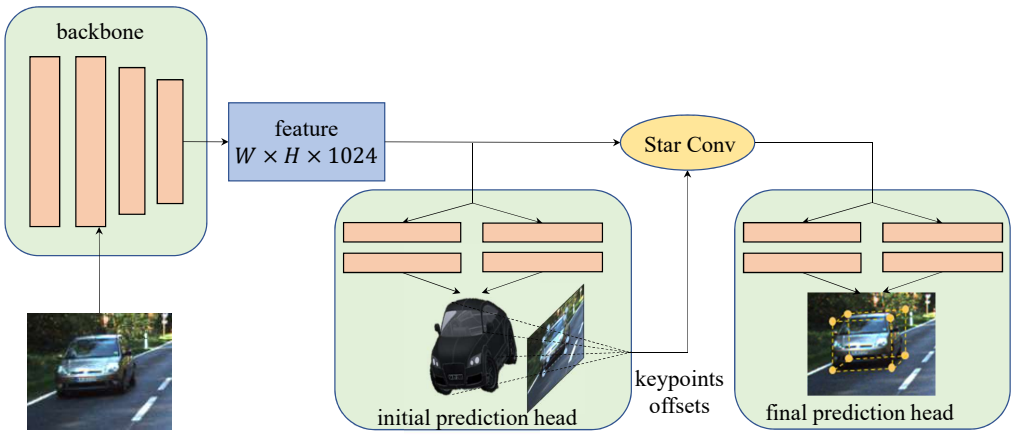

Fig. 2: The image is fed into a backbone to produce feature maps. A detection head predicts initial 3D object estimations. We then apply Star-Convolution and an additional prediction head to refine the 3D predictions.

network is still completely decoupled with the projective geometry. GAC [4], inspired by the human behavior of ground-based distance estimation, computes a depth prior from the camera's extrinsic parameters as an additional feature map for the network and samples ground-aware features from a learned offset. However, only one keypoint is sampled in the GAC module, and only a small part of the geometric information is utilized in decoding, thus limiting its ability to reason over complex geometric relations.

We propose Star-Convolution, a refinement module for image-based 3D object detection. The Star-Convolution module reprojects the initially decoded 3D objects onto the image and re-aggregates features from the sampled key points. Further, it predicts the residual between the original prediction and the ground truth. Since the decoding, projecting, and sampling operations are differentiable, the network can be trained end-to-end with calibration geometry taking effects in both inferencing and training. We test the module in both monocular and stereo setups, and we observe significant performance improvements in both cases.

The contribution of this paper is three-fold.

1) We provide a high-level view of the absence of projection geometry in the learnable inference phase of current state-of-the-art (SOTA) monocular 3D object detectors.
2) We propose the Star-Convolution refinement module for image-based 3D object detection to incorporate the decoding process in both the training and inferencing of the detector.
3) We evaluate the proposed module on the KITTI 3D object detection benchmark under both monocular and stereo settings. We show that the proposed module, though inspired by existing monocular detectors, can be extended to a more general field of visual detectors.

II. Related Works

A. Monocular 3D Object Detection

Most of the recent advances in monocular 3D object detection can be categorized into pseudo-LiDAR methods or one-stage detection methods.

Pseudo-LiDAR Methods: In pseudo-LiDAR methods, a depth prediction network first directly predict pixel-wise depth prediction rom a single RGB image. Then a 3D point-cloud-based object detector detects and localizes 3D objects based on the predicted depth [6], [7], [8], [9]. The scale-ambiguity problem is left to the depth prediction network. However, without explicit scale hints from particular objects, it is more difficult to predict depth for every pixel in the image than for just several objects of interest. Moreover, the current SOTA monocular depth prediction networks take about 0.05 s per frame, limiting the inference speed of pseudo-LiDAR methods.

One-Stage Detection Methods: One-stage detection methods are generally built upon the more mature 2D object detection architectures. Additional branches are attached to the bounding box regression branch of a 2D object detector, and these branches are supervised by the 3D parameters of the target objects. Recently, many researchers have been developing algorithms that utilize the geometry constraints between 2D bounding boxes and 3D parameters. SS3D [5] estimates 2D bounding boxes, depth, orientation, dimensions and 3D keypoints in parallel. It introduces nonlinear optimization to merge these predictions. ShiftRCNN [10] applies a sub-network to substitute the optimization, while M3D-RPN [3] introduces post-optimization steps to refine the orientation prediction. Recently, SMOKE [1] and RTM3D [11] have introduced heatmap-based keypoints prediction with an anchor-free object detector like CenterNet [12]. RTM3D also formulates the post-optimization as a nonlinear least-squares problem.

Manuscript 628 submitted to 2022 IEEE International Conference on Robotics and Automation (ICRA). Received September 14, 2021.

However, all the algorithms examined above apply mostly convolutional layers or other position-invariant operations to predict the initial 3D parameters. The network has to learn a mapping from the projective geometry under supervision, while the input and the network's inference process only contain image pixels. Ground-aware Convolution (GAC) [4] uses the calibration matrix to generate a feature map of the depth prior. To our knowledge, the proposed method is the first to embed the entire projection process inside the network and have learnable parameters **after** it.

B. Stereo 3D Object Detection

Stereo 3D object detection is considered a tractable problem, although it is computationally intensive. Several recent advances in stereo 3D object detection algorithms follow pseudo-LiDAR methods, like the monocular cases do. They apply stereo matching networks to generate the point clouds instead of applying monocular depth prediction networks. Although the detection accuracy improves significantly, the computational costs are even larger than they are when generatting a point cloud using monocular depth estimation.

Several recent advances have been achieved by limiting the computational burden and focusing the network on foreground pixels. DispRCNN [13], ZoomNet [14], and OC Stereo [15] apply instance segmentation on both images to construct a local point cloud for each proposed instance to improve the accuracy of stereo matching on foreground pixels.

RTS3D [16] and YOLOStereo3D [17], meanwhile, borrow ideas from monocular 3D object detection. They integrate stereo matching or coordinate warping modules into monocular 3D detection baselines, and achieve competitive performance with less computational cost.

We point out that advancements in monocular 3D object detection can usually be extended to stereo 3D object detection. We also test our proposed method in stereo detection tasks.

C. Bounding Box Refinement in Object Detection

In 2D object detection, multi-stage methods have been proposed to refine the accuracy of bounding boxes. Fast-RCNN [18] and Faster-RCNN [19] set up baselines for a two-stage detection structure. Algorithms like Cascade RCNN [20] further push forward the idea of refining bounding box sequentially with multi-stage detections, while Reppoints [21] applies deformable convolution [22] to refine bounding boxes in one-stage object detectors.

We propose Star-Convolution, which is inspired by 2D-based refinement algorithms, and we further utilize 3D projective geometry to refine 3D bounding box predictions under the one-stage detection frameworks.

III. Methods

In this section, we elaborate on the methods we propose in this paper.

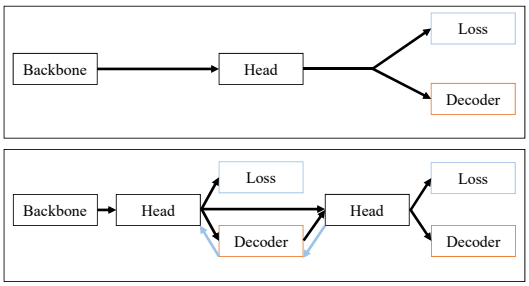

Fig. 3: The meta-architecture of the current 3D detection models (top) and the proposed model (bottom). In the bottom diagram, additional gradients propagating through the decoder are visualized with cyan arrows.

First, we give a brief recapitulation of the existing monocular image-based detection structure and GAC from a higher level. Second, we formulate Star-Convolution in 3D and explain how it exploits the projection geometry in end-to-end learning. Finally, we briefly introduce the loss function and the training scheme of the proposed module.

A. Monocular 3D Detection Nomenclature

We first recapitulate the general detection structure of multiple state-of-the-art image-based detection algorithms from a more abstract perspective, and explain the current absence of projective geometry throughout the inference process of the networks.

Although the model architectures of detectors vary, many monocular 3D detection models are built upon one-stage 2D detection models [1], [2], [3], [4], [23], [24]. Borrowing terminologies from 2D detectors [25] and re-clarifying them under the 3D setting, we extract the meta-architecture of existing image-based 3D detection models at a more abstract level. These models can be considered to consist of the following components.

1) **Backbones** Backbones are feature extractors converting images to feature maps. ResNet [26] and DLA [27] are common choices.
2) **Heads** Heads are layers that produce the designated outputs from feature maps. Heads vary depending on how the feature outputs are encoded. Most of the existing algorithms only apply basic convolutional networks in this component [1], [3], [23], [11].
3) **Losses** Losses are functions that produce supervision signals for the output of the heads. Some loss functions require complete decoding of the output [1], [5]. We note that, in most existing SOTA networks [1], [4], [17], [23], [11], the loss function does not contain learnable parameters.
4) **Decoders** Decoders are the functions that retrieve object detection outputs from the feature maps, especially during evaluation. In general, this process involves lifting the object from 2D to 3D using

TABLE I: Monocular 3D Object Detection Results of Car on KITTI Test Set

Methods	3D Easy	3D Moderate	3D Hard	BEV Easy	BEV Moderate	BEV Hard	Time
MonoPSR[9]	10.76 %	7.25 %	5.85 %	18.33 %	12.58 %	9.91 %	0.2s
PLiDAR[6]	10.76 %	7.50 %	6.10 %	21.27 %	13.92 %	11.25 %	0.1s
SS3D[5]	10.78 %	7.68 %	6.51 %	16.33 %	11.52 %	9.93 %	0.05s
M3D-RPN[3]	14.76 %	9.71 %	7.42 %	21.02 %	13.67 %	10.42 %	0.16s
RTM3D[11]	14.41 %	10.34 %	8.77 %	19.17 %	14.20 %	11.99 %	0.05s
AM3D[8]	16.50 %	10.74 %	9.52 %	25.03 %	17.32 %	**14.91** %	0.4s
D4LCN[23]	16.65 %	11.72 %	9.51 %	22.51 %	16.02 %	12.55 %	0.2s
YOLOMono3D[17]	19.24 %	12.37 %	8.67 %	27.21 %	17.24 %	12.58 %	0.05s
GAC[4]	21.65 %	13.25 %	9.91 %	**29.81 %**	17.98 %	13.08 %	0.05s
Ours	**21.97 %**	**14.06 %**	**10.01 %**	**28.14 %**	**18.67 %**	**13.79** %	**0.06s**

the camera model, and does *not* contain learnable parameters.

The inference structure and the gradient flow of current monocular 3D models are visualized in the upper part of Figure 3. Note that the camera model is only used in the decoder, and it does not provide supervising feedback to the network during the training process. Therefore, the current state-of-the-art detection networks adapt to a particular camera configuration to produce distance estimation of the objects with position-invariant convolutional networks. Projective geometries are absent during the learning phase.

B. Overview of Ground-Aware Convolution

GAC is a module designed for monocular 3D object detection and depth prediction that extracts features and depth prior information from the ground plane pixels [4]. It is a significant improvement on the "head" of the structure. The concept of GAC is visualized in Figure 1 (a).

Given a feature map $\mathcal{F}$ as the input, the GAC module first computes a depth/pseudo-disparity prior, assuming all the pixels are on the ground plane, and attaches this prior to the original feature map. Then, for a pixel (x, y) in the feature map, we first estimate an offset Δy from the current pixel to the ground plane. Finally, we sample the features from the predicted ground plane pixel $(x, y + \Delta y)$ and aggregate them with the features at (x, y).

The GAC module explicitly injects projection geometry into the monocular 3D object detection pipeline and significantly improves the detection accuracy. However, it is clear that there are some further short-comings of the GAC module:

1) It only captures features at a single ground plane pixel, which may not be sufficient for complex geometric reasoning.
2) The network has to learn an offset from the center pixel to the ground plane without explicit supervision. The learned offset is ambiguous for the network.

These issues hinder the module's capability to more accurately capture features from other geometric key points that are potentially more informative.

C. Star-Convolution in 3D

To fully exploit the 3D geometric information in image-based 3D object detection, we further extend the concepts in the GAC module with a 3D Star-Convolution.

Inspired by the Star-DConvolution in 2D [29], which is a bounding box refinement module that extracts features from the sides of 2D bounding boxes, we propose to apply Star-Convolution to extract features from the projected keypoints of the initially detected objects, as indicated in Figure 1 (b) and Figure 1 (c).

From a pixel (x, y) in the feature map, a 3D object detector will predict one (anchor-free detector) or multiple (anchor-based detector) objects. We first select the 3D box prediction with the highest confidence for each pixel. These objects are encoded as $X_0 = (cx, cy, z, w, h, l, \alpha)$, where z, (w, h, l), α and (cx, cy) is the distance, dimensions, orientation and the projection of its 3D center, respectively. The results are then further lifted to 3D to form an initial guess of a 3D object $(x_{3d}, y_{3d}, z, w, h, l, \theta)$ with the calibration matrix P. Then, we project the corner points of this rectangular 3D object onto the image. The equation for the first keypoint is provided as follows as an example:

$$\begin{bmatrix} x_{kp_0} \\ y_{kp_0} \end{bmatrix} = \tilde{P} \begin{bmatrix} \frac{x_{3d} + 0.5 \cdot (l \cos\theta + w \sin\theta)}{z_{3d} + 0.5 \cdot (-l \sin\theta + w \cos\theta)} \\ \frac{y_{3d} + 0.5 \cdot h}{z_{3d} + 0.5 \cdot (-l \sin\theta + w \cos\theta)} \\ 0 \\ 1 \end{bmatrix}, \tag{1}$$

where $\tilde{P}$ is the first two rows of the calibration matrix.

We apply deformable convolution [22] to extract features from nine sampling points, which are chosen as the eight projected keypoints $\{(x_{\mathrm{kp}_i}, y_{\mathrm{kp}_i}) | i \in [1, 8]\}$ and the current pixel point (x, y). The projection process and the bilinear sampling operation are differentiable.

Intuitively, instead of sampling from a single ground pixel from a learned offset as done in GAC, Star-Convolution utilizes the results from the decoders to extract features and more depth priors from the eight projected keypoints. Moreover, the prior 2D/3D information in the anchors is also incorporated into the inference process.

Manuscript 628 submitted to 2022 IEEE International Conference on Robotics and Automation (ICRA). Received September 14, 2021.

TABLE II: Stereo 3D Oject Detection Results on the KITTI Test Set.

Methods	3D Easy	3D Moderate	3D Hard	BEV Easy	BEV Moderate	BEV Hard	Time
RT3DStereo[24]	29.90 %	23.28 %	18.96 %	58.81 %	46.82 %	38.38 %	0.08s
StereoRCNN[28]	47.58 %	30.23 %	23.72 %	61.92 %	41.31 %	33.42 %	0.30s
Pseudo-LiDAR*[6]	54.53 %	34.05 %	28.25 %	67.30 %	45.00 %	38.40 %	0.40s
OC Stereo*[15]	55.15 %	37.60 %	30.25 %	68.89 %	51.47 %	42.97 %	0.35s
ZoomNet*[14]	55.98 %	38.64 %	30.97 %	72.94 %	54.91 %	44.14 %	0.35s
YOLOStereo3D [17]	65.68 %	41.25 %	30.42 %	76.10 %	50.28 %	36.86 %	0.08s
Ours	**67.32 %**	**42.56 %**	**30.82 %**	**77.20 %**	**51.34 %**	**36.74 %**	**0.08s**

The extracted features are further fed into convolutional layers to predict a residual $\Delta X = (\Delta cx, \Delta cy, \Delta z, \Delta w, \Delta h, \Delta l, \Delta \alpha)$. These refinement layers do not share weights with the base output layers. The final output of the module is simply the aggregation: $X' = X_0 + \Delta X$.

The gradient flow of the network structure enhanced with Star-Convolution is visualized at the bottom of Figure 3. A more detailed visualization of the inference process is presented in Figure 2. We point out that the network can now be trained with supervising feedback from camera projections, and the refinement head can produce more accurate detection predictions.

In the proposed Star-Convolution pipeline, there are learnable modules **following** a full decoder; thus, these modules can learn to predict residuals by examining the projected key points of the initial guess during training and inferencing. Such a characteristic differentiates the proposed module from those in existing works where camera projection is only used in constructing loss functions or decoding final outputs.

D. Learning Scheme and Loss Function

The proposed network has two detection output branches, one from the baseline model and the other from the Star-Convolution refinement. Both are trained with supervision from the ground truth label.

During backpropagation, since all the operations are differentiable, the gradients will also flow through the decoding and projection process, improving the output quality of the refinement module. The model is thus trained end-to-end.

To supervise the prediction of 3D boxes, we apply focal loss [30], [31] on classification, and smoothed-L1 loss [18] on bounding box regression. The total loss $\mathcal{L}$ is the sum of the classification loss L_{cls} and regression loss L_{reg} for the two branches:

$$\mathcal{L} = \alpha(L_{cls_0} + L_{reg_0}) + (L'_{cls} + L'_{reg})$$

where the parameter α is a balancing weight. We will give a more detailed discussion of the parameter α in Section V.

IV. Experiments

A. Experiments Settings and Training Details

1) Dataset: We perform the experiments on the KITTI dataset [32], which contains 7,481 training frames and 7,518 test frames. For choosing hyperparameters and conducting ablation studies, we further split the training set into 3,712 training frames and 3,769 validation frames, following Chen [33].

2) Model: Our experiments on monocular 3D object detection are based on the YOLOMono3D baseline [17]. We add two extra convolutional layers to predict the residual between the ground-truth and the first prediction from the refined feature. The enhanced monocular model runs at about 16 frames per second (FPS).

We note that YoloStereo3D [17] also shares the inference structure of a monocular 3D detector. Thus, we also enhance it with the Star-Convolution refinement head. The enhanced stereo model runs at about nine FPS, which is much faster than other SOTA stereo algorithms.

3) Training Details: We follow the training scheme of in [4] and [17]. Both the left and right RGB image data are utilized in training the monocular detectors [4], [17], [11]. The top 100 pixels are cropped to speed up inference. After a simple grid search on the validation set , we select $\alpha = 0.3$ as the baseline parameter choice for the loss function.

4) Evaluation Protocol: The evaluation protocol closely follows other SOTA methods [4], [17] the official KITTI benchmark [32]. We evaluate performance with 40 recall positions (AP_{40}) instead of the 11 recall positions (AP_{11}) proposed in the original Pascal VOC benchmark[34]. All presented results on the test set and also ablation study based on AP_{40}.

B. Performance Increment from Star-Convolution

The results for monocular 3D object detection are presented in Table I. The proposed methods outperforms most of the current state-of-the-art monocular methods. We further point out that the proposed Star-Convolution also outperforms the baseline YOLOMono3D.

Qualitative results from the monocular 3D object detectors are shown in Figure 4. The figure demonstrates that our proposed Star-Convolution module can improve the original prediction results and improve 3D detection accuracy.

The results for stereo 3D object detection are presented in Table II, where a similar performance boost from the baseline methods can be observed. Because most of the stereo detection's accuracy depends on the

Manuscript 628 submitted to 2022 IEEE International Conference on Robotics and Automation (ICRA). Received September 14, 2021.

Fig. 4: Visualizetion, in both image view and bird-eye-view, of some qualitative results. Green boxes are final predictions of the network, blue boxes are predictions from the base branches, and red boxes are ground truth annotations.

quality of stereo matching, the relative improvement is less than that of the monocular methods.

TABLE III: Monocular 3D Detection Ablation Study Results of Car on KITTI Validation Set

Methods	$IoU \geq 0.7$ **3D Easy/Moderate/Hard**
Baseline Model	**23.12 %/ 15.64 %/ 11.81 %**
YOLOMono3D [17]	21.66 %/ 14.20 %/ 11.07 %
2D Star-Conv	22.31 %/ 15.10 %/ 11.26 %
Deformable Conv	22.38 %/ 14.64 %/ 11.50 %
$\alpha = 0.0$	19.89 %/ 12.87 %/ 10.02 %
$\alpha = 0.5$	23.52 %/ 15.28 %/ 11.28 %
$\alpha = 1.0$	22.72 %/ 14.86 %/ 11.55 %

V. Ablation Study and Discussion

In this section, we take a closer look into the proposed Star-Convolution with multiple experiments.

A. Star-Convolution vs. Additional Heads

We first conduct experiments on the design of the Star-Convolution. In the experiments, we first substitute 3D star convolution with the 2D star convolution proposed in [29], while the refinement paradiam remained the same.

The Star-Convolution we propose also enlarges the receptive fields of the network. We conduct another experiment using standard deformable convolutional layers to substitute for the convolutional head of YOLOMono3D. The results are presented in Table III. We empirically show that, while both deformable convolution and 2D Star-Convolution can improve the detection performance, 3D Star-Convolution outperforms them both.

We also note that the result on the validation set is lower than the reported number in GAC [4] while we outperforms it in test set, which is because the proposed Star Convultion is more robust to the variation in the extrinsic/intrinsic paramter of the camera.

B. Parameter α

The hyper-parameter α weights the relative importance between the two prediction branches and is an important parameter determining the behavior of the proposed Star-Convolution module. We present the experiment results by varying the hyper-parameter α within $[0, 1]$.

The results are presented in Table III. In the case of $\alpha = 0.0$, the initial bounding box proposal for Star-Convolution is not trained and is far from the correct objects; thus, the performance decreases instead. If α becomes too large, the network may fall back to the un-refined baseline, and the performance will drop. Although the choice of α seems critical, the detection accuracy is stable between $\alpha = 0.3$ and $\alpha = 0.5$. We show that the performance is comparatively insensitive to the choice of α in this range.

VI. Conclusion

In this paper, we presented Star-Convolution for image-based 3D object detection. First, we briefly reviewed the state-of-the-art monocular 3D object detectors from an abstract level, and identified the absence of projection geometry in the training phase of the network. Second, we introduced Star-Convolution as a refinement module for both monocular and stereo 3D object detectors. The proposed module wraps the projection process inside both the inference and end-to-end training processes. We tested both the monocular and stereo enhanced networks on the KITTI detection benchmark and achieved SOTA performance among purely vision-based methods. Finally, we conducted further experiments to demonstrate that the Star-Convolution refinement module is indeed a superior design to other naive modules.

Although the proposed module increases the computation burden of current image-based detection models, we manage to further push the performance potential of 3D object detection algorithms. We produce more powerful neural network models for autonomous driving and vision-based mobile robots.

Manuscript 628 submitted to 2022 IEEE International Conference on Robotics and Automation (ICRA). Received September 14, 2021.

REFERENCES

[1] Zechen Liu, Zizhang Wu, and Roland Tóth. SMOKE: Single-stage monocular 3d object detection via keypoint estimation. *arXiv preprint arXiv:2002.10111*, 2020.

[2] Yongjian Chen, Lei Tai, Kai Sun, and Mingyang Li. Monopair: Monocular 3d object detection using pairwise spatial relationships. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[3] Garrick Brazil and Xiaoming Liu. M3D-RPN: monocular 3d region proposal network for object detection. *CoRR*, abs/1907.06038, 2019.

[4] Y. Liu, Y. Yuan, and M. Liu. Ground-aware monocular 3d object detection for autonomous driving. *IEEE Robotics and Automation Letters*, 2021.

[5] Eskil Jörgensen, Christopher Zach, and Fredrik Kahl. Monocular 3d object detection and box fitting trained end-to-end using intersection-over-union loss. *CoRR*, abs/1906.08070, 2019.

[6] Xinshuo Weng and Kris Kitani. Monocular 3d object detection with pseudo-lidar point cloud. *CoRR*, abs/1903.09847, 2019.

[7] Jean Marie Uwabeza Vianney, Shubhra Aich, and Bingbing Liu. Refinedmpl: Refined monocular pseudolidar for 3d object detection in autonomous driving, 11 2019.

[8] Xinzhu Ma and/ Zhihui Wang, Haojie Li, Wanli Ouyang, and Pengbo Zhang. Accurate monocular 3d object detection via color-embedded 3d reconstruction for autonomous driving. *CoRR*, abs/1903.11444, 2019.

[9] Jason Ku, Alex D. Pon, and Steven L. Waslander. Monocular 3d object detection leveraging accurate proposals and shape reconstruction. *CoRR*, abs/1904.01690, 2019.

[10] Andretti Naiden, Vlad Paunescu, Gyeongmo Kim, ByeongMoon Jeon, and Marius Leordeanu. Shift R-CNN: deep monocular 3d object detection with closed-form geometric constraints. *CoRR*, abs/1905.09970, 2019.

[11] Pei-Xuan Li, Huaici Zhao, Pengfei Liu, and Feidao Cao. Rtm3d: Real-time monocular 3d detection from object keypoints for autonomous driving. *ArXiv*, abs/2001.03343, 2020.

[12] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. In *arXiv preprint arXiv:1904.07850*, 2019.

[13] Jiaming Sun, Linghao Chen, Yiming Xie, Siyu Zhang, Qinhong Jiang, and and Hujun Bao Xiaowei Zhou. Disp r-cnn: Stereo 3d object detection via shape prior guided instance disparity estimation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[14] Zhenbo Xu, Wei Zhang, Xiaoqing Ye, Xiao Tan, Wei Yang, Shilei Wen, Errui Ding, Ajin Meng, and Liusheng Huang. Zoomnet: Part-aware adaptive zooming neural network for 3d object detection. *arXiv preprint arXiv:2003.00529*, 2020.

[15] Alex D. Pon, Jason Ku, Chengyao Li, and Steven L. Waslander. Object-centric stereo matching for 3d object detection. *arXiv preprint arXiv:1909.07566*, 2019.

[16] Peixuan Li, Shun Su, and Huaici Zhao. Rts3d: Real-time stereo 3d detection from 4d feature-consistency embedding space for autonomous driving. In *arXiv preprint arXiv:2012.15072*, 2020.

[17] Yuxuan Liu and Ming Liu. Yolostereo3d: A step back to 2d for efficient stereo 3ddetection. In *arXiv preprint arXiv:2102.15072*, 2021.

[18] Ross B. Girshick. Fast R-CNN. *CoRR*, abs/1504.08083, 2015.

[19] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6):1137–1149, 2017.

[20] Z. Cai and N. Vasconcelos. Cascade r-cnn: High quality object detection and instance segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2019.

[21] Z. Yang, S. Liu, H. Hu, L. Wang, and S. Lin. Reppoints: Point set representation for object detection. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9656–9665, 2019.

[22] Xizhou Zhu, Han Hu, Stephen Lin, and Jifeng Dai. Deformable convnets v2: More deformable, better results. *CoRR*, abs/1811.11168, 2018.

[23] Mingyu Ding, Yuqi Huo, Hongwei Yi, Zhe Wang, Jianping Shi, Zhiwu Lu, and Ping Luo. Learning depth-guided convolutions for monocular 3d object detection. 12 2019.

[24] Konigshof Hendrik, Salscheider Niels, and Stiller Christoph. Realtime 3d object detection for automated driving using stereo vision and semantic information. pages 1405–1410, 10 2019.

[25] Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiarui Xu, Zheng Zhang, Dazhi Cheng, Chenchen Zhu, Tianheng Cheng, Qijie Zhao, Buyu Li, Xin Lu, Rui Zhu, Yue Wu, Jifeng Dai, Jingdong Wang, Jianping Shi, Wanli Ouyang, Chen Change Loy, and Dahua Lin. MMDetection: Open mmlab detection toolbox and benchmark. *arXiv preprint arXiv:1906.07155*, 2019.

[26] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.

[27] F. Yu, D. Wang, E. Shelhamer, and T. Darrell. Deep layer aggregation. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2403–2412, 2018.

[28] Peiliang Li, Xiaozhi Chen, and Shaojie Shen. Stereo R-CNN based 3d object detection for autonomous driving. *CoRR*, abs/1902.09738, 2019.

[29] Haoyang Zhang, Ying Wang, Feras Dayoub, and Niko Sünderhauf. Varifocalnet: An iou-aware dense object detector. *arXiv preprint arXiv:2008.13367*, 2020.

[30] Peng Yun, Lei Tai, Yuan Wang, and Ming Liu. Focal loss in 3d object detection. *CoRR*, abs/1809.06065, 2018.

[31] Tsung-Yi Lin, Priyal Goyal, Ross Girshick, Kaiming He, and Piotr Dollar. Focal loss for dense object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PP:1–1, 07 2018.

[32] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[33] Xiaozhi Chen, Kaustav Kundu, Yukun Zhu, Andrew G Berneshawi, Huimin Ma, Sanja Fidler, and Raquel Urtasun. 3d object proposals for accurate object class detection. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 424–432. Curran Associates, Inc., 2015.

[34] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, June 2010.