SIDE: Center-based Stereo 3D Detector with Structure-aware Instance Depth Estimation

Xidong Peng[1], Xinge Zhu[2], Tai Wang[2], and Yuexin Ma[1]

[1]ShanghaiTech University
[2]The Chinese University of Hong Kong

{linmo1533, zhuxinge123, taiwang.me}@gmail.com, mayuexin@shanghaitech.edu.cn

Abstract

3D detection plays an indispensable role in environment perception. Due to the high cost of commonly used LiDAR sensor, stereo vision based 3D detection, as an economical yet effective setting, attracts more attention recently. For these approaches based on 2D images, accurate depth information is the key to achieve 3D detection, and most existing methods resort to a preliminary stage for depth estimation. They mainly focus on the global depth and neglect the property of depth information in this specific task, namely, sparsity and locality, where exactly accurate depth is only needed for these 3D bounding boxes. Motivated by this finding, we propose a stereo-image based anchor-free 3D detection method, called structure-aware stereo 3D detector (termed as SIDE), where we explore the instance-level depth information via constructing the cost volume from RoIs of each object. Due to the information sparsity of local cost volume, we further introduce match reweighting and structure-aware attention, to make the depth information more concentrated. Experiments conducted on the KITTI dataset show that our method achieves the state-of-the-art performance compared to existing methods without depth map supervision.

1. Introduction

3D object detection is important for scene understanding and widely applied in many applications, such as autonomous driving [25, 40, 24] and virtual reality. The rapid progress of 3D detectors have been witnessed in recent years and most state-of-the-art approaches leverage the data collected by LiDAR [18, 33, 4, 22, 15] considering it can provide accurate 3D information. However, LiDAR is expensive to deploy or maintain in practical use, and has limited sensing range in some cases, which makes the vision-
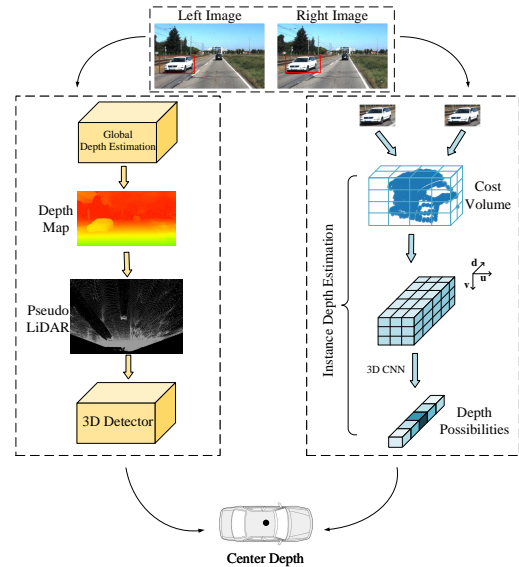

Figure 1: The left shows global depth estimation used in previous work while the right is instance depth estimation used in our work. Our depth estimation is only for objects' center and does not need depth map for supervision.

based 3D detection methods draw more attention recently. Compared to LiDAR-based methods, the key information about depth is not directly manifest in the given 2D images. This problem is especially prominent for monocular cases [25, 16, 27, 2, 28]. In comparison, binocular methods could leverage the stereo geometry to physically compute the depth, and thus are more reasonable approaches to detecting 3D objects from 2D images. Therefore, we target at the key problem of depth estimation and aim at proposing an efficient and effective stereo-based 3D detector.

Previous stereo-based work [38, 41, 32] typically estimates the global depth field to assist subsequent 3D de-

tection. Although it achieves promising performance even close to LiDAR-based methods, there are several limitations. First, it is time-consuming due to the computational overheads introduced by per-pixel dense estimation. In addition, considering most of the area in the image belongs to the background, a better performance of global estimation sometimes needs a trade-off from foreground regions, which are actually more important in this specific task. Furthermore, it needs the dense depth labels for supervision, and thus brings extra annotation costs. Hence, instead of estimating the depth globally, focusing on the depth estimation of instances with only 3D bounding boxes labels can be more effective in stereo-based 3D detection.

In this paper, we propose a novel 3D detection method, *i.e.*, SIDE, to solve above problems, in which it performs depth estimation only for objects' center with corresponding 2D regions of interest (RoI). Compared with previous methods, the dense depth labels are not required and running time can be reduced with the simplification of required depth estimation. Fig.1 shows the comparison between our work and previous work intuitively. During depth estimation, we first introduce match-reweight strategy to take advantage of the internal similarity of cost volume. Since instance depth information is sparse in space and the position of object's 3D center is not directly represented in the image, we introduce structure-aware attention mechanism to extract the structural information of local patch in the front view and the bird's eye view by convolution, then condense the information into the original cost volume to make the depth feature more concentrated. Based on the accurate depth estimation, we further propose a simple and efficient post-processing under the geometric constraint to refine our detection results.

We evaluate the proposed SIDE method with KITTI 3D dataset [10]. Specifically, our AP_{3D} of car category is better than the state-of-the-art methods IDA-3D [30] in all kinds of cases with IoU=0.7. Especially, in the moderate and hard case, our method performs better than IDA-3D with over 3% AP_{3D}, which means our method can better detect objects far away or with large occlusions.

The contributions of our proposed method are mainly summarized as follows.

1) We investigate the natural property of depth information in stereo 3D detection, namely, sparsity and locality, and reroute the global depth to structure-aware instance depth.

2) We introduce a novel stereo 3D detector by accurately predicting the center depth of each object with the match-reweight and structure-aware attention.

3) A simple yet effective post-processing method is proposed to refine the detection results under the geometric constraint.

4) Evaluated on the KITTI 3D dataset, we achieve state-of-the-art performance compared with the stereo-based methods without depth map supervision.

2. Related Work

LiDAR based methods   Most of the state-of-the-art 3D detection methods rely on LiDAR, because it provides accurate depth information of objects. These methods process LiDAR data in different representations. [8, 17, 44, 45, 35, 46, 47] utilize structured voxel representation to quantize the LiDAR data and feed them into 2D or 3D CNN to detect 3D object, while [4, 22, 15] project the LiDAR data into 2D bird's eye view or front view representations. Instead of transforming the representations of point cloud, [31, 33] directly takes raw point cloud as input to localize 3D object based on the frustum region. Additionally, the idea of anchor-free is applied to the LiDAR method in [9, 34], which reduces the time consuming for 3D detection. Although the performance of LIDAR-based methods is superior, compared with the high cost of LIDAR, the image-based methods are more practical currently.

Moncular image based methods   Because monocular cameras are cheaper than LiDAR or stereo cameras, monocular-based 3D object detection naturally becomes a hot spot for both industry and academia. [25, 16, 27, 36, 37] extend the state-of-the-art 2D object detector to regress the orientation and dimensions of the object's 3D bounding box. [2, 28] explicitly utilize sparse information by predicting series of keypoints of regular-shape vehicles, then the 3D object pose can be constrained by wireframe template fitting. [21] predicts the nine perspective key points of the 3D bounding box, and uses geometric constraint to recover the three-dimensional information of the object. [6] considers the relationship between paired samples to improve monocular 3D target detection. These methods are cheap and fast, but the performance is not satisfying because it is difficult to obtain accurate depth information, which is very critical for 3D detection.

Stereo image based methods   Stereo cameras are much cheaper than LiDAR and stereo images can provide depth information implicitly through the disparity, which make it attract more and more attentions. Stereo-based 3D detection methods extract the implicit depth information of stereo images in different ways. [3] focuses on encoding object size prior, ground-plane prior, and depth information into an energy function to generate 3D proposals. [19] converts the 3D object detection problem to left and right 2D object detection and keypoint prediction, then uses geometric constraints to build the 3D detection box. [38, 41, 32] convert the estimated disparity map of the stereo image into pseudo LiDAR points, then use LiDAR-based methods to estimate the three-dimensional bounding boxes.   These methods
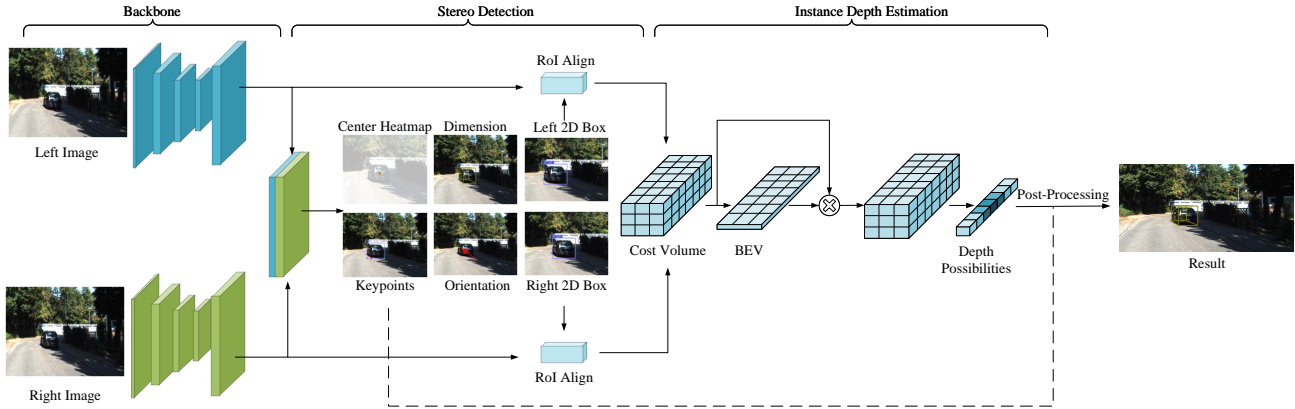
Figure 2: Network architecture of the proposed SIDE. It takes only the stereo RGB images as inputs. After the backbone network, the network outputs 7 branches which predict center depth, 2D center heatmap, stereo 2D boxes, object's kepoints, dimension and steering angle of 3D box based on the idea of anchor-free. Center depth branch is different from others, in which we perform instance depth estimation by selecting 2D RoI of objects. In order to get accurate depth, we introduce match-reweight and structure-aware attention in the depth estimation to make depth feature more focused. Finally, detection results are further refined through the post-processing.

achieve the most advanced performance on the stereo image but cannot detect in real time because it is time-consuming to estimate the depth of entire image. Some traditional methods [26, 39] use region-based stereo matching to estimate object's depth, but their performance is poor because they are not based on deep learning. [30] predicts the depth of the target through the method of instance depth perception which can detect the three-dimensional box end-to-end, but it neglects the local structure information. Our approach introduces a structure-aware-depth-estimation module that directly predicts the depth of the 3D bounding box's center, and then rectifies the results by box estimation and dense alignment, which together benefit the accuracy of depth estimation and thus yield better 3D detection performance.

3. SIDE

3.1. Overview

Given the input stereo RGB images, our goal is to predict 3D bounding boxes and category labels for each object of interest. The attributes of predicted bounding box include the position of the object center (x, y, z), the three-dimensional size (w, h, l), and the steering angle θ. Our method only needs the labels of 3D bounding boxes as supervision for each image.

The complete framework is shown in Fig.2. For stereo detection, we associate the position of objects in the left and right images through the heatmap of objects' center. The details will be introduced in Section 3.2. For instance depth estimation, we only pay attention to the depth information of the center point of each object, so we construct

local cost volume based on the RoIs of objects. Since the target depth information has strong sparsity and locality in the local cost volume, we make use of the structure and internal similarity of local cost volume to aggregate features. As a result, match reweight and structure-aware attention are introduced to make the information more concentrated and thus enhance the accuracy of depth estimation. The details will be introduced in Section 3.3. Finally, given the preliminary accurate estimated depth, we devise an efficient geometric post-processing scheme with the 3D-2D projection formula to further correct objects' positions. The details are in the Section 3.4.

3.2. Stereo Detection

Stereo 2D Detection Similar to [43], we use the heatmap of object's center to represent each object. The heatmap is generated through the backbone and other branches are linked according to the position of each object in the heatmap. According to correspondence, the respective bounding boxes of each object in the stereo pictures can be detected at the same time.

Usually, object's position is corrected in stereo pictures to ensure the vertical position of the same object is the same, so $y1$ and $y2$ in the bounding box of the stereo image are the same and we only need to predict a shared y when predicting the position. Therefore, the detection of the object's stereo 2D box is completed by the heatmap and two other branches, whose predictions are (o_{ul}, o_{ur}, o_v), (w_l, w_r, h). As shown in Fig.3, o_{ul} and o_{ur} represent the respective offset from the horizontal u coordinate of the object's center in the heatmap to stereo images, and o_v represents the offset
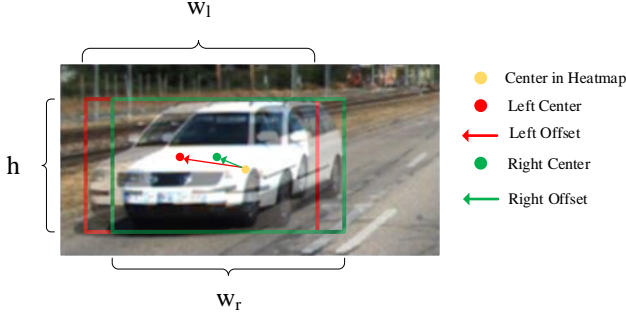
Figure 3: The relationship between left image and right image through center heatmap.

of vertical v coordinate. w_r and w_l represent the respective widths of object's stereo 2D box, h represents the same height of the 2D box.

For the generation of the image center heatmap, we use the same method in CenterNet [43] and set $Y \in [0, 1]^{\frac{W}{R} \times \frac{H}{R} \times C}$ to label the image, where W represents the width of image, H represents the height of image, R represents the multiple of downsampling, and C represents the type of classification. In order to alleviate the imbalance problem of positive and negative samples, we use focal loss [23] to train this network. Since the network downsampled the image by R, when the feature map is remapped to the original image, it will cause errors. Therefore, an additional offset needs to be predicted for each center and we predict the offset from the object's center in the heatmap to its center in the left and right image. In addition, we regress the length and width of the 2D stereo detection boxes after downsampling which are calculated in advance. All of these values are trained with L1 loss.

Three-dimensional Attributes Detection In order to construct the 3D bounding box of the object, we need extra branches to predict the three-dimensional attributes of each object. Therefore, we also make predictions for the object's dimensions, steering angle, and keypoints. The dimensions include the length, width and height of 3D box, and keypoints include object's perspective keypoints [19] and visible keypoints. Perspective keypoints used for constructing 3D-2D projection can help correct the steering angle and visible keypoints used for further dense alignment can help correct the location of 3D object in the post-processing.

For the prediction of object's dimensions, because 3D dimensions of an object are three scalars, We directly regress to their absolute values in meters using a separate head. For the prediction of the steering angle θ of the object, we predict the allocentric angle α of each object instead of directly predicting the egocentric steering angle θ. The two angles can be transformed by $\theta = \alpha + \arctan(\frac{x}{z})$. To avoid the discontinuity, the training targets are $[\sin\alpha, \cos\alpha]$ instead of

the raw angle value. both values are trained with L1 Loss.

3.3. Structure-Aware Instance Depth Estimation

Depth estimation is a very challenging problem in 3D detection. Next, we will elaborate the details of our devised structure-aware instance depth estimation module for tackling this problem in stereo-based 3D detection. Most previous work[38, 41, 32] construct cost volume of entire picture to calculate the disparity relationship corresponding to every pixel, then transform it to get the entire picture's depth information, which requires the use of depth map or disparity map for supervision during the training process. Compared with the previous, our depth estimation only pays attention to the information of the center depth of each detected object, and only needs to estimate the center depth of the object. Therefore, the cost volume needed to be constructed is smaller, and only the annotated object depth information in the 3D ground truth is required for supervision.

The detail of structure-aware instance depth estimation module is shown in Fig.4. We use RoI Align[11] to select the area of target object on the feature map of the stereo images, build 4D cost volume from the selected feature, and then feed it into a 2-stage 3D convolutional network. After reweighting the cost volume by calculating the similarity of the feature area, it will be fed into the first-stage 3D convolutional network, where we introduce a structure-aware attention mechanism to make the information more concentrated. In the second stage, we use Max Pooling to perform 2 times downsampling after passing 3D convolutional network twice respectively, then we use Max Pooling to perform 4 times downsampling and SoftMax to normalize the prediction. Relying on the network's normalization, the down-sampled features are finally merged into depth probability of the 3D box center. Therefore, the final result of the object's center depth can be calculated by $\hat{z} = \sum_{i=1}^{N} z_i \times P(i)$, where N denotes the number of depth levels and $P(i)$ is the normalized probability. We train our model with supervised learning using ground truth depth of 3D box center, where supervised regression loss is defined using the error between the ground truth depth z and the model's predicted depth $\hat{z}$

$$L_{depth} = \frac{1}{N} \sum_{k=1}^{N} |z^{(k)} - \hat{z}^{(k)}|. \tag{1}$$

Construction of Cost Volume We use the depth as the dividing basis to ensure that the depth range is evenly divided, then convert the uniform depth range into the non-uniform disparity range, and construct the cost volume with the non-uniform disparity range. Disparity d can be converted to depth z by $z = \frac{f_u \times b}{d}$, where f_u represents horizontal focal length and b represents the baseline of stereo camera. This equation shows that disparity and depth are in an inverse relationship, which means if the disparity is directly used as
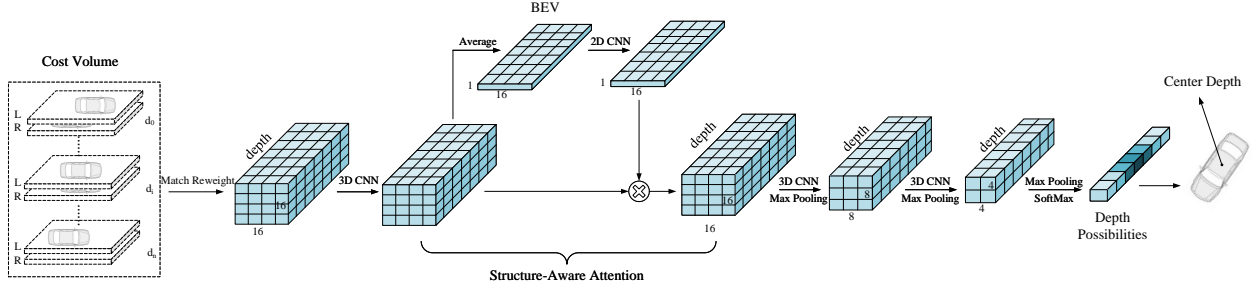
Figure 4: Depth estimation module builds a 4D cost volume and performs 3DCNN to estimate the depth of a 3D bounding box center. We use two modules Match Reweight and Structure-Aware Attention to make information more focused and thus beneficial for estimating the depth.

the dividing basis of cost volume, the far area will be insufficiently divided leading to inaccurate depth estimation of distant objects. Our construction of cost volume based on depth ensure we are also more likely to obtain accurate depth about objects at far location.

In addition, since we estimate the depth for each specific object individually, so we do not need to construct cost volume in such a large depth range. According to the width of the 2D box in stereo image, we can roughly calculate a more precise depth range for depth estimation by the camera intrinsic parameters.

Match Reweight Strategy When calculating the predicted depth, we will perform a weighted average according to the normalized possibility of all depths in the cost volume range to get final result, rather than the depth value represented by the maximum probability of the cost volume, which ensures the continuity of the depth estimation range. However, when the difference in depth possibility is not obvious, the depth result will lack discriminativeness. Therefore, when constructing the cost volume, we introduced correlation scores to reweight the cost volume sequence. As shown in Eq.2, the correlation score s at depth level i is obtained by calculating the correlation between left and right feature maps in cost volume, where $F_l^{(i)}$ and $F_r^{(i)}$ are the pair of feature maps at corresponding depth level i.

$$s^{(i)} = \cos < F_l^{(i)}, F_r^{(i)} >= \frac{F_l^{(i)} \cdot F_r^{(i)}}{||F_l^{(i)}|| \times ||F_r^{(i)}||}. \quad (2)$$

This equation uses cosine function to calculate the similarity of the corresponding regions of the feature map in the cost volume of each level. Then we use this similarity to reweight corresponding depth level in the cost volume. Our reweight method mainly comes from the idea that, in the construction of cost volume at different depth levels, the contents contained in the corresponding selected feature maps are different. When the similarity between the

two feature maps is high, it means that there are more corresponding regions in the two feature maps at this depth level, which also means that this level is more likely to represent the center depth of the object.

Structure-aware Attention Mechanism Although the match reweight strategy makes the cost volume's depth level more discriminant, there is also lots of spatial information noise in the 4D information space due to the sparsity of local cost volume. When we estimate the depth, we only estimate the depth of the center point of the object, which means the whole information space of cost volume is not well used. Inspired by some LiDAR-based methods[4, 22, 15], which convert the intermediate feature space to the perspective of front view or bird's-eye-view to reduce the interference of unstructured spatial noise, we design a structure-aware attention module for the stereo image based 3D detection.

As shown in the structure-aware attention part of Fig.4, after feeding the local cost volume into a 3D convolutional network, we averaged it on the Y-axis to get a depth feature space in 2D bird's-eye-view, then we use a 2D convolution and Sigmoid function σ to determine which parts in the feature space of bird's-eye-view are useful. Finally, the convolution result is multiplied by the original cost volume to achieve the effect of reducing space noise. The attention process can be expressed by Eq.3

$$G_a = \sigma(Conv(Avg(G_h, dim = 2))) \otimes G_h + G_h \quad (3)$$

3.4. Refinement of 3D Posture

After obtaining object's 2D box, 3D dimension, steering angle, and the depth of object's center, object's 3D position can be roughly calculated. Subsequently, we combine the objects' predicted perspective keypoints and visual keypoints to further correct the 3D position. In the postprocessing, we use box estimation and dense alignment to get more accurate results.

5

Geometric 3D Box Estimation According to [19, 20], given the left-right 2D boxes, perspective keypoint, and regressed dimensions, the 3D box center (x, y, z, θ) can be solved by minimizing the projection error of 2D boxes and the keypoint. Since our network has a depth estimation module to get an accurate depth, there is no need to estimate the depth through the box estimator. Therefore, our box estimator is simpler than the box estimator adopted by Stereo R-CNN, we only need to use the 2D box information of the left picture, combined with the preliminary predicted depth z and the keypoints to correct the data of x, y and θ. the projection formula of 3D-2D we formed is shown in Eq.4. We extract five measurements from 2D boxes and perspective keypoints $(u_l, v_t, u_r, v_b, u_p)$, which represent left, top, right, bottom edges of the left 2D box, and the u coordinate of the perspective keypoint. w, h and l represent the regression size, and x, y, z represent the coordinates of the center point of the 3D bounding box.

$$\begin{cases} u_l = (x - \frac{w}{2}\cos\theta - \frac{l}{2}\sin\theta) / (z + \frac{w}{2}\sin\theta - \frac{l}{2}\cos\theta) \\ v_t = (y - \frac{l}{2}) / (z + \frac{w}{2}\sin\theta - \frac{l}{2}\cos\theta) \\ u_r = (x + \frac{w}{2}\cos\theta + \frac{l}{2}\sin\theta) / (z - \frac{w}{2}\sin\theta + \frac{l}{2}\cos\theta) \\ v_b = (y + \frac{l}{2}) / (z - \frac{w}{2}\sin\theta + \frac{l}{2}\cos\theta) \\ u_p = (x + \frac{w}{2}\cos\theta - \frac{l}{2}\sin\theta) / (z - \frac{w}{2}\sin\theta - \frac{l}{2}\cos\theta) \end{cases}$$
(4)

Note that with regard to the perspective keypoints, only one of the bottom corners can be projected within the vertical border of the 2D box visually and the keypoints observed in different perspectives about the same car are different. Therefore, when predicting key points, we not only predict the distance of the keypoint relative to the border, but also predict the type of keypoint. In addition, when the object is truncated on the image, We use the viewpoint angle α to compensate the unobservable states.

Dense 3D Box Alignment Within visual range from the predicted visual keypoints, we can sample the pixel of the object from the stereo images and calculate the corresponding error of each pixel at a given center depth, then we can obtain a more accurate depth by minimizing the sum of pixel's error. In practice, as shown in Fig.5, we find that the prediction of the visual range about some highly occluded objects will be inaccurate, resulting in the center depth of the object after dense alignment is even more inaccurate, so we improve the original dense alignment module. For objects with a large occlusion, the visible range will be further reduced to ensure that the sampled points are from the object's 3D box, not from other objects that cover it. In addition, since the depth estimation can predict relatively accurate depth information, the depth range in the dense alignment can be reduced to speed up this module.
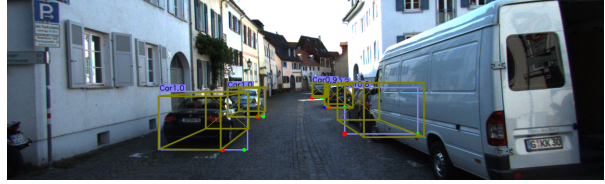

Figure 5: The red dot represents the left boundary of the visible range, and the green dot represents the right boundary of the visible range. It can be seen from the figure that the prediction of the visible range is inaccurate for cars with severe occlusion.

4. Experiments

Backbone Like the implementation in [43], we use dla-34 [42] as the backbone of the network, and all hierarchical aggregation connections are replaced by DCN [7]. The number of channels output by dla34 is 64 and five of the detection branches are connected to the backbone network through two convolutions of sizes $3 \times 3 \times 64$ and $1 \times 1 \times n$, where n is the characteristic channel of the relevant output branch. For the detection of keypoints, since it has more output, we deepen this branch to get more accurate results. As for the branch of the depth prediction, we first reduce the dimension of the stereo feature map through a convolution, then construct local cost volume to predict the center depth of each object through the instance depth estimation.

Training We implement the models in PyTorch [29] and define the loss function of this multitask training by Eq. 5. Each loss is weighted by their uncertainty following [13]. In addition, We double the training set by flipping the training set image horizontally and swap left and right images. We also adopts random clip and scale data augmentation for training and set the probability of random clip and scale to 0.35 to prevent excessive information from overflowing the image. We use the Adam [14] optimizer on 4 NVIDIA Tesla V100 GPU for 80 epochs of training. The initial learning rate is set to 2.5×10^{-4}. The learning rate is reduced by 10 times at 25th, 40th, 60th and 70th epoch respectively. The backbone network is initialized by a classification model pretrained on ImageNet [12]. we train with 16 batch sizes of each GPU for about 9 hours.

$$\begin{aligned} L = &w_{cls}L_{cls} + w_{off}L_{off} + w_{size}L_{size} + w_{dim}L_{dim} \\ &+ w_{\theta}L_{\theta} + w_{kpts}L_{kpts} + w_{depth}L_{depth}. \end{aligned}$$
(5)

Evaluation We evaluate our method on the KITTI object detection benchmark [10]. Following the same training and validation splits as [3], we devide 3712 images into traing set and 3769 images into validation set respectively. We report 3D average precision AP_{3D} and birds-eye-view aver-

Table 1: Average precision of bird's eye view AP_{BEV} and 3D boxes AP_{3D}, evaluated on the KITTI validation set, where S denotes stereo image as input, M denotes monocular image as input and D denotes extra depth map as supervision.

Method	Data	Time	AP_{BEV} / AP_{3D}(IoU=0.5)			AP_{BEV} / AP_{3D}(IoU=0.7)		
			Easy	Mode	Hard	Easy	Mode	Hard
M3D-RPN [1]	M	160ms	55.37/38.96	42.49/39.57	35.29/33.01	25.97/20.27	21.18/17.06	17.09/15.21
MonoPair [6]	M	57ms	61.06/55.38	47.63/42.39	41.92/37.99	24.12/16.28	18.17/12.30	15.76/10.42
RTM3D [21]	M	55ms	57.47/54.36	44.16/41.90	42.31/35.84	25.56/20.77	22.12/16.86	20.91/16.63
PL+FP [38]	S+D	670ms	89.80/89.50	77.60/75.50	68.20/66.30	72.80/59.40	51.80/39.80	44.00/33.50
PL+AVOD [38]	S+D	510ms	88.50/76.80	76.40/65.10	61.20/56.60	61.90/60.70	45.30/39.20	39.00/37.00
PL++ [41]	S+D	500ms	89.00/89.00	77.50/77.80	68.70/69.10	74.90/63.20	56.80/46.80	49.00/39.80
DSGN [5]	S+D	-	-	-	-	83.24/72.31	63.91/54.27	57.83/47.71
3DOP [33]	S	-	55.04/46.04	41.25/34.63	34.55/30.09	12.63/6.55	9.49/5.07	7.59/4.10
S-RCNN [19]	S	417ms	87.13/85.84	74.11/66.28	58.93/57.24	68.50/54.11	48.30/36.69	41.47/31.07
IDA-3D [30]	S	-	88.05/87.08	**76.69/74.57**	67.29/60.01	70.68/54.97	50.21/37.45	42.93/32.23
SIDE(R=4)	S	210ms	86.41/85.29	74.43/67.21	66.45/59.05	68.75/56.74	51.21/41.83	44.97/35.67
SIDE(R=2)	S	260ms	**88.35/87.70**	76.01/69.13	**67.46/60.05**	**72.75/61.22**	**53.71/44.46**	**46.16/37.15**

age precision AP_{BEV} on car category with the IoU thresholds at 0.5 and 0.7. The category of car is divided into easy, moderate, and hard case according to the 2D box height, occlusion and truncation levels.

4.1. 3D Detection Performance on KITTI

We conduct experiments both qualitatively and quantitatively. For comparison, we summarize the results mainly into two groups, monocular-based and stereo-based methods, then we set the downsampling factor R = 2 and R = 4 respectively to train the model and evaluate the performance of our 3D detection method by Average Precision for bird's eye view AP_{BEV} and 3D box AP_{3D} as shown in Tab.1.

Compared with stereo-based methods without depth map supervision, we obtain the highest AP_{3D} and AP_{BEV}. Specifically, we outperform 3DOP over 30% for both AP_{BEV} and AP_{3D} across all kinds of cases. The performance of our model is also much better than Stereo R-CNN in the AP_{BEV} and AP_{3D} because Stereo R-CNN calculates the center depth of the object through a geometric method instead of directly predicting the center depth of the object, which causes inaccurate depth prediction. For IDA-3D, since this method will estimate the depth of object center, its performance is also better than Stereo R-CNN. But our method introduces the match reweight and attention mechanism to make the information more aggregated as well as uses geometric and pixel-level constraints to refine detection results in the post-processing, so the performance of our method are also comparable with IDA-3D, and it is better than IDA-3D in the easy and hard case with IoU=0.5 and all kinds of cases with IoU=0.7.

We also list some stereo-based methods that require

Table 2: The performance of 3D detection on the KITTI testing set

Method	AP_{3D}(IoU=0.7)		
	Easy	Mode	Hard
Stereo-RCNN	47.67	30.23	23.72
SIDE	**47.69**	**30.82**	**25.68**

depth map as supervision. Although some of them perform better than our method, they all run much slower than our method. Compared with previous monocular-based methods, although the running time of these methods is lower than ours, our method outperforms previous monocular-based methods by a significant margin in all kinds of cases. These comparisons show that our method achieves a good balance between performance and efficiency. Furthermore, we report evaluation results on the KITTI testing set in Tab.2, compared with the testing set results of Stereo-RCNN, our method also shows superiority especially on the hard class.

4.2. Ablation Study

We conduct some ablation experiments to show the contribution of our proposed network modules.These experiments are performed on the car category in the KITTI dataset and we set the downsampling factor R = 4.
Depth Estimation and Post-processing We first conduct experiments to verify the effect of instance depth estimation and geometric post-processing on the performance of 3D detection. The experimental results are shown in Tab.3. When the instance depth estimation is not used, we directly use the center point disparity of the stereo 2D detection box

Table 3: Contribution of instance depth estimation and geometric post-processing

config	AP_{BEV} / AP_{3D}(IoU=0.5)		
	Easy	Mode	Hard
w/o Estimation	54.24/32.02	41.84/26.77	39.95/26.82
w/ Estimation	66.73/32.15	58.20/31.68	51.90/28.96
w/ Estimation w/ Post-processing	**86.41/85.29**	**74.43/67.21**	**66.45/69.05**

to calculate the center depth, which leads to inaccurate 3D detection results. After adding instance depth estimation module, it shows that the performance of 3D detection especially in AP_{BEV} has been greatly improved, and the final performance can be further improved through geometric post-processing.

Match Reweight and Attention Module During depth estimation, correlation score reweights cost volume according to the similarity of different depth levels and structure-aware attention reduces the noise in the original space by the information from bird's eye view of the object. The purpose of these two modules is to improve the possibility of correct depth and make the result of depth estimation more discriminant. As is shown in Tab. 4, the performance of our method can be improved by combining these two modules.

Table 4: Improvements of match-reweight and structure-aware attention, where Re. represents match reweight and Att. represents structure-aware attention.

Att.	Re.	AP_{BEV} / AP_{3D} (IoU=0.5)		
		Easy	Mode	Hard
		84.39/83.45	72.93/65.71	65.08/57.78
✓		85.84/84.84	73.88/66.91	66.02/58.80
	✓	85.46/84.63	73.95/66.81	66.28/58.86
✓	✓	**86.41/85.29**	**74.43/67.21**	**66.46/59.04**

Box Estimator and Dense Alignment Although the coarse 3D box has a precise projection on the image because of the depth estimation, it is not accurate enough for 3D localization. Therefore, we need to correct the steering angle of 3D box through box estimation and refine the depth of 3D box through dense alignment. The results in Tab.5 show that the performance of our method is improved through these two steps. Note that in the process of dense alignment, we need to use box estimation again after fixing the depth to recover the 3D box. In addition, we find that the improvement of 3D detection is limited when only dense alignment is used, because it requires 3D boxes to fit closely with objects, while the box estimator can make these 3D boxes fit more closely with corresponding objects by correcting the steering angle.

The Benefit of Trainig Strategy We use two strategies to

Table 5: Improvements of dense alignment and 3D box estimation, where Est. represents 3D box estimation and Ali. represents dense alignment. .

Est.	Ali.	AP_{BEV} / AP_{3D} (IoU=0.5)		
		Easy	Mode	Hard
		66.73/32.15	58.20/31.68	51.90/28.06
✓		81.35/74.61	64.29/61.93	56.81/54.41
	✓	74.88/69.46	62.00/58.30	54.83/51.85
✓	✓	**86.41/85.29**	**74.43/67.21**	**66.45/59.05**

enhance model performance during the training stage, image flip augmentation and weight uncertainty. We conduct different combinations of experiments on these two strategies and the results are shown in Tab. 6. Weight uncertainty can balance the multi-task loss and avoid manual adjustment of weights. The data flip augmentation doubles the number of samples to achieve better accuracy. Both of the strategies make our method achieve better performance.

Table 6: Improvements of using data flip augmentations and uncertainty weight.

Flip	Uncertainty	AP_{BEV} / AP_{3D} (IoU=0.5)		
		Easy	Mode	Hard
		77.16/75.52	62.48/60.32	55.11/53.29
✓		79.23/77.42	64.53/62.40	57.03/55.31
	✓	80.13/77.19	63.79/61.90	56.42/54.38
✓	✓	**86.41/85.29**	**74.43/67.21**	**66.45/59.05**

Qualitative Results We show the qualitative results of some scenarios from KITTI dataset in the supplementary materials.

5. Conclusion

In this paper, we propose a center-based stereo 3D detection method which has better performance especially when detecting objects in hard condition (such as farther from the camera and more severe occlusion). Since depth information is essential for 3D detection, we estimate object's center depth via constructing local cost volume from its RoI. We also introduce match-reweight and structure-aware attention to aggregate information and reduce space noise caused by information sparsity. By overcoming this sparsity, the center of object can be predicted more accurately. In addition, we use the anchor-free model to speed up objects' 2D detection and further improve the accuracy of results in the post-processing through object's geometric and pixel-wise constraint. By predicting more accurate location of keypoints, the corrected result can be more accurate.

References

[1] Garrick Brazil and Xiaoming Liu. M3d-rpn: Monocular 3d region proposal network for object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9287–9296, 2019.

[2] Florian Chabot, Mohamed Chaouch, Jaonary Rabarisoa, Céline Teuliere, and Thierry Chateau. Deep manta: A coarse-to-fine many-task network for joint 2d and 3d vehicle analysis from monocular image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2040–2049, 2017.

[3] Xiaozhi Chen, Kaustav Kundu, Yukun Zhu, Huimin Ma, Sanja Fidler, and Raquel Urtasun. 3d object proposals using stereo imagery for accurate object class detection. *IEEE transactions on pattern analysis and machine intelligence*, 40(5):1259–1272, 2017.

[4] Xiaozhi Chen, Huimin Ma, Ji Wan, Bo Li, and Tian Xia. Multi-view 3d object detection network for autonomous driving. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1907–1915, 2017.

[5] Y Chen, S. Liu, X. Shen, and J. Jia. Dsgn: Deep stereo geometry network for 3d object detection. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[6] Yongjian Chen, Lei Tai, Kai Sun, and Mingyang Li. Monopair: Monocular 3d object detection using pairwise spatial relationships. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12093–12102, 2020.

[7] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 764–773, 2017.

[8] Martin Engelcke, Dushyant Rao, Dominic Zeng Wang, Chi Hay Tong, and Ingmar Posner. Vote3deep: Fast object detection in 3d point clouds using efficient convolutional neural networks. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1355–1361. IEEE, 2017.

[9] Runzhou Ge, Zhuangzhuang Ding, Yihan Hu, Yu Wang, Sijia Chen, Li Huang, and Yuan Li. Afdet: Anchor free one stage 3d object detection. *arXiv preprint arXiv:2006.12671*, 2020.

[10] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3354–3361. IEEE, 2012.

[11] K. He, G. Gkioxari, P Dollár, and R. Girshick. Mask r-cnn. In *IEEE*, 2017.

[12] D. Jia, D. Wei, R. Socher, L. J. Li, L. Kai, and F. F. Li. Imagenet: A large-scale hierarchical image database. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[13] Alex Kendall, Yarin Gal, and Roberto Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7482–7491, 2018.

[14] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[15] Jason Ku, Melissa Mozifian, Jungwook Lee, Ali Harakeh, and Steven L Waslander. Joint 3d proposal generation and object detection from view aggregation. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1–8. IEEE, 2018.

[16] Jason Ku, Alex D Pon, and Steven L Waslander. Monocular 3d object detection leveraging accurate proposals and shape reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11867–11876, 2019.

[17] Bo Li. 3d fully convolutional network for vehicle detection in point cloud. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1513–1518. IEEE, 2017.

[18] Bo Li, Tianlei Zhang, and Tian Xia. Vehicle detection from 3d lidar using fully convolutional network. *arXiv preprint arXiv:1608.07916*, 2016.

[19] Peiliang Li, Xiaozhi Chen, and Shaojie Shen. Stereo r-cnn based 3d object detection for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7644–7652, 2019.

[20] P. Li, T. Qin, and S. Shen. Stereo vision-based semantic 3d object and ego-motion tracking for autonomous driving. *Springer, Cham*, 2018.

[21] Peixuan Li, Huaici Zhao, Pengfei Liu, and Feidao Cao. Rtm3d: Real-time monocular 3d detection from object keypoints for autonomous driving. *arXiv preprint arXiv:2001.03343*, 2, 2020.

[22] Ming Liang, Bin Yang, Shenlong Wang, and Raquel Urtasun. Deep continuous fusion for multi-sensor 3d object detection. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 641–656, 2018.

[23] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017.

[24] Yuexin Ma, Xinge Zhu, Sibo Zhang, Ruigang Yang, Wenping Wang, and Dinesh Manocha. Trafficpredict: Trajectory prediction for heterogeneous traffic-agents. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6120–6127, 2019.

[25] Fabian Manhardt, Wadim Kehl, and Adrien Gaidon. Roi-10d: Monocular lifting of 2d detection to 6d pose and metric shape. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2069–2078, 2019.

[26] Suresh B Marapane and Mohan M Trivedi. Region-based stereo analysis for robotic applications. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(6):1447–1464, 1989.

[27] Arsalan Mousavian, Dragomir Anguelov, John Flynn, and Jana Kosecka. 3d bounding box estimation using deep learning and geometry. In *Proceedings of the IEEE Conference*

on Computer Vision and Pattern Recognition, pages 7074–7082, 2017.

[28] J Krishna Murthy, GV Sai Krishna, Falak Chhaya, and K Madhava Krishna. Reconstructing vehicles from a single image: Shape priors for road scene understanding. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 724–731. IEEE, 2017.

[29] A. Paszke, S. Gross, F. Massa, A. Lerer, and S. Chintala. Pytorch: An imperative style, high-performance deep learning library. 2019.

[30] Wanli Peng, Hao Pan, He Liu, and Yi Sun. Ida-3d: Instance-depth-aware 3d object detection from stereo vision for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13015–13024, 2020.

[31] Charles R Qi, Wei Liu, Chenxia Wu, Hao Su, and Leonidas J Guibas. Frustum pointnets for 3d object detection from rgb-d data. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 918–927, 2018.

[32] Rui Qian, Divyansh Garg, Yan Wang, Yurong You, Serge Belongie, Bharath Hariharan, Mark Campbell, Kilian Q Weinberger, and Wei-Lun Chao. End-to-end pseudo-lidar for image-based 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5881–5890, 2020.

[33] Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. Pointrcnn: 3d object proposal generation and detection from point cloud. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 770–779, 2019.

[34] Guojun Wang, Bin Tian, Yunfeng Ai, Tong Xu, Long Chen, and Dongpu Cao. Centernet3d: An anchor free object detector for autonomous driving. *arXiv preprint arXiv:2007.07214*, 2020.

[35] Tai Wang, Xinge Zhu, and Dahua Lin. Reconfigurable voxels: A new representation for lidar-based point clouds. In *Conference on Robot Learning*, 2020.

[36] Tai Wang, Xinge Zhu, Jiangmiao Pang, and Dahua Lin. FCOS3D: Fully convolutional one-stage monocular 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, 2021.

[37] Tai Wang, Xinge Zhu, Jiangmiao Pang, and Dahua Lin. Probabilistic and geometric depth: Detecting objects in perspective. *arXiv preprint arXiv:2107.14160*, 2021.

[38] Yan Wang, Wei-Lun Chao, Divyansh Garg, Bharath Hariharan, Mark Campbell, and Kilian Q Weinberger. Pseudo-lidar from visual depth estimation: Bridging the gap in 3d object detection for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8445–8453, 2019.

[39] Zeng-Fu Wang and Zhi-Gang Zheng. A region based stereo matching algorithm using cooperative optimization. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2008.

[40] S. Yang and S. Scherer. Cubeslam: Monocular 3d object slam. 2018.

[41] Yurong You, Yan Wang, Wei-Lun Chao, Divyansh Garg, Geoff Pleiss, Bharath Hariharan, Mark Campbell, and Kilian Q Weinberger. Pseudo-lidar++: Accurate depth for 3d object detection in autonomous driving. *arXiv preprint arXiv:1906.06310*, 2019.

[42] F. Yu, D. Wang, E. Shelhamer, and T. Darrell. Deep layer aggregation. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

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

[44] Yin Zhou and Oncel Tuzel. Voxelnet: End-to-end learning for point cloud based 3d object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4490–4499, 2018.

[45] Xinge Zhu, Yuexin Ma, Tai Wang, Yan Xu, Jianping Shi, and Dahua Lin. Ssn: Shape signature networks for multi-class object detection from point clouds. In *Proceedings of the European Conference on Computer Vision*, 2020.

[46] Xinge Zhu, Hui Zhou, Tai Wang, Fangzhou Hong, Wei Li, Yuexin Ma, Hongsheng Li, Ruigang Yang, and Dahua Lin. Cylindrical and asymmetrical 3d convolution networks for lidar-based perception. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[47] Xinge Zhu, Hui Zhou, Tai Wang, Fangzhou Hong, Yuexin Ma, Wei Li, Hongsheng Li, and Dahua Lin. Cylindrical and asymmetrical 3d convolution networks for lidar segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9939–9948, 2021.