Homography Loss for Monocular 3D Object Detection

Jiaqi Gu[1,2], Bojian Wu[1]*, Lubin Fan[1], Jianqiang Huang[1], Shen Cao[1], Zhiyu Xiang[2], Xian-Sheng Hua[1]
[1]Alibaba Cloud Computing Ltd., [2]Zhejiang University

Abstract

Monocular 3D object detection is an essential task in autonomous driving. However, most current methods consider each 3D object in the scene as an independent training sample, while ignoring their inherent geometric relations, thus inevitably resulting in a lack of leveraging spatial constraints. In this paper, we propose a novel method that takes all the objects into consideration and explores their mutual relationships to help better estimate the 3D boxes. Moreover, since 2D detection is more reliable currently, we also investigate how to use the detected 2D boxes as guidance to globally constrain the optimization of the corresponding predicted 3D boxes. To this end, a differentiable loss function, termed as Homography Loss, is proposed to achieve the goal, which exploits both 2D and 3D information, aiming at balancing the positional relationships between different objects by global constraints, so as to obtain more accurately predicted 3D boxes. Thanks to the concise design, our loss function is universal and can be plugged into any mature monocular 3D detector, while significantly boosting the performance over their baseline. Experiments demonstrate that our method yields the best performance (Nov. 2021) compared with the other state-of-the-arts by a large margin on KITTI 3D datasets.

1. Introduction

Monocular 3D object detection is a fundamental task in computer vision, where the goal is to localize and estimate 3D bounding boxes, parameterized by location, dimension, and orientation, of objects from a single image. It can be applied to various scenes, such as autonomous driving, robotic navigation, etc. However, it is an ill-posed and challenging problem since a single image cannot provide explicit depth information. To acquire such resources, most existing methods resort to LiDAR sensors to obtain accurate depth measurements [29], or stereo cameras for stereo depth estimation [15], but they will increase the cost of practical usages. In comparison, the monocular camera is cost-effective.

Figure 1. (a) Most of existing methods consider each object as a single training sample, (b) our proposed *homography loss* establishes connections between objects, and applies 2D detection as guidance to help constrain 3D localization in (c) Bird's Eye View.

Most of the existing monocular 3D object detection methods have already achieved remarkable high accuracy with fixed camera settings. However, in their training strategies, each 3D object in the scene is treated as an individual sample without considering the mutual relationships with other neighboring objects, for example, as shown in Fig. 1(a). Assuming that, if the predicted 3D box of a single object obviously deviates from its ground truth, without additional constraints, it is usually hard for the network to refine and correct the estimated position of this specific sample. To handle this, apart from the regression loss defined by minimizing the discrepancies between the predicted 3D boxes and the ground truths, many algorithms propose projection loss [15, 17, 25, 26] to constrain the optimization of 3D boxes with the supervision of corresponding projected 2D ground truth boxes. However, the 3D localization of a single object is still independent of the others. Differently, MonoPair [7] exploits the object relationships and builds scene graph to enhance the mutual connections of objects during training and inference. They fully leverage the spatial relationships between close-by objects instead of individually focusing on the information-constrained single object. An obvious drawback is that an object can only *locally* connect with its nearest neighbor.

On the other hand, a large percent of approaches are effective for normal objects. In reality, only the foreground

---

*Corresponding author: ustcbjwu@gmail.com

objects can be detected easily, because they are fully visible and have rich recognizable features. Therefore, these approaches still struggle to handle the occluded objects or small ones that are far away from the camera, and those objects usually occupy a higher proportion in the scene. Limited improvement is achieved since little information is helpful to solve the problem. A straightforward way to improve the 3D detection is to correct the results by the foreground objects or even the 2D detection results. The most relevant work, MonoFlex [42], which leverages the distribution of different objects and proposes a flexible framework to decouple the truncated objects and adaptively combine multiple approaches for 3D detection. However, it is also limited to training the network for each individual sample.

Moreover, due to the perspective projection, objects with different depths may block each other in image space. Thus, OFTNet [33] and ImVoxelNet [34] propose to regress 3D positions on Bird's Eye View (BEV), since objects on the projected BEV plane do not intersect with each other and can be distinguished.

In general, to be concrete as shown in Fig. 1, our core idea is to build the connections between all the objects and globally optimize their 3D positions. Besides, we also associate BEV with image view through inverse projective mapping and apply 2D detection results as guidance to improve the 3D localization in BEV. To achieve the goal, we propose *Homography Loss* to combine 2D and 3D information and globally balance the mutual relationships to obtain more accurate 3D boxes. By doing so, our loss function is able to effectively encode necessary geometric information in both 2D and 3D space, and the network will be enforced to explicitly capture the global geometric relationships between objects which are demonstrated to be helpful for 3D detection. Because of the differentiability and interpretability, our loss function can be plugged into any mature monocular 3D detector. Practically, we take ImVoxelNet [34] and MonoFlex [42] as examples, and integrate the novel homography loss during training phase, experiments demonstrate that our method outperforms the state-of-the-arts by a large margin on KITTI 3D detection benchmark (Nov. 2021). The main contributions can be summarized as follows:

- We propose a novel loss function, termed as *homography loss*, to exploit geometric relationships of all the objects in the scene and globally constrain their mutual locations, by using the homography between the image view and the Bird's Eye View. At the same time, the geometric consistency in both 2D and 3D space will be well preserved. To the best of our knowledge, this is the first work that fully leverages the *global* geometric constraints in monocular 3D object detection.

- The proposed monocular 3D detector based on homography loss achieves the state-of-the-art performance on KITTI 3D detection benchmark, and surpasses the results of all the other monocular 3D detectors, which implies the superiority of our loss.

- We apply this loss function to several popular monocular 3D detectors. Without any additional inference cost, the training is more stable and easier to converge, achieving higher accuracy and performance. It can be a plug-and-play module and be adapted to any monocular 3D detector.

2. Related Work

We first review methods on monocular 3D object detection, followed by a brief introduction of geometric constraints that are commonly used during training phase.

Monocular 3D object detection is an ill-posed problem because of lacking depth clues of the monocular 2D image. When compared with stereo images [15] or LiDAR-based methods [23, 27, 30, 32, 39, 40], in some earlier works, auxiliary information are necessary for monocular 3D detection to achieve competitive results. These prior knowledge usually includes ground plane assumption [5], morphable wireframe model hypothesis [13] or 3D CAD model [3, 14], etc.

Moreover, some other works only take a single RGB image as input. For example, Deep3DBox [25] estimates the 3D pose and dimension from the image patch enclosed by a 2D box. Afterwards, the network with a 3D regression head [9, 18, 26] is used to predict the 3D box while searching and filtering the proposal whose 2D projection has the threshold overlap with the ground-truth 2D box. MonoGRNet [31] detects and localizes 3D boxes via geometric reasoning in both the observed 2D projection and the unobserved depth dimension. MonoDIS [36] leverages a novel disentangling transformation for 2D and 3D detection losses. M3D-RPN [1] reformulates the monocular 3D detection problem as a standalone 3D region proposal network. Unlike previous methods, which depend on 2D proposals, SMOKE [19] argues that the 2D detection network is redundant and introduces non-negligible noise in 3D detection. Thus, it predicts a 3D box for each object by combining a single keypoint estimated with regressed 3D variables via a single-stage detector, and similarly, RTM-3D [17] predicts nine perspective keypoints of a 3D box in the image space. Specifically, MonoFlex [42] proposes a flexible framework for monocular 3D object detection that explicitly decouples the truncated objects and adaptively combines multiple approaches for depth estimation.

However, image-based training and inference will introduce non-linear perspective distortion where the scale of objects varies drastically with depth, which makes it hard to accurately predict the relative distance and location of the object of interest. To handle this, OFTNet [33] proposes orthographic feature transform by mapping image-based fea-
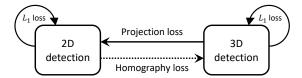
Figure 2. In practice, for 2D/3D detection tasks, the discrepancies between the predicted 2D/3D boxes and the corresponding ground truths can be narrowed down by applying L1 loss. It means that the predicted 2D/3D boxes would be self-constrained with the corresponding ground truths. Besides, the predicted 3D positions of objects can be projected into 2D image space with camera parameters, and the projected 2D positions will be further compared with their 2D ground truths. That is to say, 3D data can be converted to 2D space via a projection matrix. By analogy, our proposed method builds the correlation from 2D to 3D and uses 2D detection as guidance to supervise the training of 3D localization.
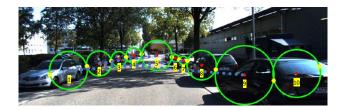

Figure 3. The location of the target object is globally affected by other objects. Since an individual object can only locally connect with its nearest neighbor, as proposed in [7], the pairwise relationship is not enough to encode the spatial relationship of objects. We take the global affection into account, which is similar to the long-range dependency used in the attention mechanism. For example, the location of Car 2 can not only be influenced by Car 1, but also be constrained by Car 5 and 9 as connected with the blue dotted line. (The figure originates from [7])

tures into an orthographic 3D space that is better aligned with the real-world perception, where the target object will not intersect or occlude with each other and can be intuitively distinguished. ImVoxelNet [34] projects the obtained image features extracted from the backbone network to a 3D voxel volume and proposes to detect 3D boxes from BEV for the same purpose as just mentioned.

Overall, current methods consider either directly regressing depth or keypoints from image view, or detecting 3D boxes from BEV. As none of the existing methods dig into the inherent connection between the image view and BEV, our proposed method first bridges the gap between them.

Geometric constraints in 3D detection. Most current approaches directly regress 3D spatial information from 2D image without the help of extra 3D priors. Because 2D and 3D space are naturally interrelated via perspective projection, therefore, some recent works attempt to use geometric constraints in the network. Mousavian *et al*. [25] estimates 3D boxes using the geometric relations between 2D edges and 3D corners. Li *et al*. [15] solves a coarse 3D box by utilizing the sparse perspective keypoints and 2D box. Naiden *et al*. [26] solves the translation vector of the object center via a closed-form least squares equation. Li *et al*. [17] utilizes the geometric relationship of 3D and 2D perspectives to recover 3D boxes. Li *et al*. [16] reformulates the non-linear optimization in the projective space as a differentiable geometric reasoning module. Note that, the aforementioned methods apply the geometric constraint to individual object. Contrarily, we take the positional relationships of all the objects into account at the same time.

3. Methods

3.1. Motivation

We have two key observations: 1) the 2D detection can serve as a guidance to constrain and supervise the training of

3D localization, 2) the position of a single object should be *globally* influenced by the surrounding objects, as detailed in Fig. 2 and 3. To handle those problems, we propose *homography loss* to implement the conversion from 2D image space to 3D BEV space, and simultaneously constrain the globally geometric relationships of all the objects.

3.2. Revisiting of Homography

A homography is a mapping between two planar surfaces which preserves collinearity. The homography matrix $\mathbf{H} \in \mathbb{R}^{3\times3}$ between two 2D planes maps $\mathbf{p}_1$ in the plane 1 to $\mathbf{p}_2$ in the plane 2 up to a scale factor s. It satisfies:

$$s\mathbf{p}_2 = \mathbf{H}\mathbf{p}_1, \tag{1}$$

where $\mathbf{p} = [x, y, 1]^T$ is the homogeneous coordinate of a 2D point in a plane. Since the homography matrix has 8 degrees of freedom, at least 4 corresponding point pairs are necessary for recovering the matrix. Inspired by ImVoxelNet [34], the projections of objects on BEV plane do not intersect with each other and accordingly contain more information about 3D localization, we define the homography matrix between the image plane and BEV plane, in order to implicitly transform coordinates from 2D to 3D space. More details will be illustrated in Sec. 3.3. Then, let us explain why homography is a global geometric constraint. Firstly, all pairs of corresponding points will involve in solving the homography matrix from Eq. 1, and the solution is guaranteed to be globally optimal. In other words, the constraint enforced by arbitrary pair of corresponding points will finally affect the whole optimization process. Thus, homography is a global constraint. Secondly, in projective geometry, a homography is an isomorphism of projective spaces, which correlates a group of points on one plane to the other and preserves geometric properties, e.g., collinearity. So, homography is also a geometric constraint.

Figure 4. 2D and 3D candidate points of a single object.

3.3. Homography Loss

Inspired by those observations, we propose a global loss function, termed as *homography loss*, aiming to establish the geometric connections among all the objects by leveraging the homography matrix. Assuming that we already have a monocular 3D object detector that could predict 3D boxes under the supervision of the ground truths, in addition to the regular classification and regression loss in the common pipelines, our homography loss penalizes the wrong relationship among all the predicted boxes and refines the final locations. The major steps are listed as follows.

Candidate Points Modeling. Suppose we have the predicted boxes box_{pred} obtained from the arbitrary 3D detector and the corresponding ground truth boxes box_{gt}. As mentioned in Sec. 3.2, we opt to use the homography matrix to describe the projection relationship between the image plane and the BEV plane. For a single object, as demonstrated in Fig. 4, we pick up five bottom points $\mathbf{Q}_{pred} = [x_{pred}, y_{pred}, z_{pred}]^T$ of box_{pred} as representatives, including one bottom center point and four bottom corner points. We also assume that all the objects are always on the flat ground, the bottom points on the BEV plane can thus be simplified as $\tilde{\mathbf{Q}}_{pred} = [x_{pred}, y_{pred}]^T$. Similarly, we have $\mathbf{Q}_{gt} = [x_{gt}, y_{gt}, z_{gt}]^T$ obtained from box_{gt}. After the camera projection, the ground truth 3D box will be transformed into the image space, which is defined by:

$$\mathbf{q} = \mathbf{K}\left[\mathbf{R}|\mathbf{t}\right]\mathbf{Q}, \tag{2}$$

where $\mathbf{K}$ is the intrinsic matrix and $[\mathbf{R}|\mathbf{t}]$ are the extrinsic matrices, and $\mathbf{q} = [u, v]^T$ represents the projected pixel on the image plane, which is suitable for both box_{pred} and box_{gt}. Therefore, if there exist N objects, we can get $5N$ pairs of candidate points $\mathbf{q}_{pred}, \tilde{\mathbf{Q}}_{pred}$ for box_{pred} and $\mathbf{q}_{gt}, \tilde{\mathbf{Q}}_{gt}$ for box_{gt}, respectively, which are prepared for calculating the homography matrix.

Calculating Homography. To implicitly constrain relative positions of each object, without loss of generality, we select $\mathbf{q}_{gt}$ and $\tilde{\mathbf{Q}}_{pred}$. Specifically, we use the ground truth coordinates $\mathbf{q}_{gt}$ in 2D image view as guidance, to correct the final positions $\tilde{\mathbf{Q}}_{pred}$ in 3D space. The formulation is defined, up to a scale factor (omitted here) with homoge-

neous coordinates, as follows,

$$\tilde{\mathbf{Q}}_{pred} = \mathbf{H}\mathbf{q}_{gt}, \begin{pmatrix} x_{pred} \\ y_{pred} \\ 1 \end{pmatrix} = \mathbf{H}\begin{pmatrix} u_{gt} \\ v_{gt} \\ 1 \end{pmatrix}. \tag{3}$$

Here, $\mathbf{H}$ stores the mutual relationships of all the objects by mapping between two views. We use singular value decomposition (SVD) to calculate the homography matrix $\mathbf{H}$ as it can be easily implemented in PyTorch [28].

In practice, the homography matrix in Eq. 3 is estimated since $\tilde{\mathbf{Q}}_{pred}$ may deviate a lot from the ground truth at the very beginning of training. We denote it as $\hat{\mathbf{H}}$, and represent $\tilde{\mathbf{Q}}_{homo} = \hat{\mathbf{H}}\mathbf{q}_{gt}$. As the training progresses, the estimated value $\tilde{\mathbf{Q}}_{homo}$ will approach $\tilde{\mathbf{Q}}_{pred}$ and $\tilde{\mathbf{Q}}_{gt}$.

Loss Function. The homography matrix $\hat{\mathbf{H}}$ implicitly contains the correspondences between two different views and the relative positions of all the objects. Previously, 3D detection is treated as an independent task for each object, which is constrained by regression loss, such as $L_{reg} = \text{L1}\left(\tilde{\mathbf{Q}}_{\text{gt}} - \tilde{\mathbf{Q}}_{\text{pred}}\right)$. Here, we propose a novel loss function, named as *homography loss*, to optimize the locations with strong spatial constraints. The homography loss is defined as follows,

$$\begin{aligned} L_{homo} &= \text{SmoothL1}\left(\tilde{\mathbf{Q}}_{\text{gt}} - \tilde{\mathbf{Q}}_{\text{homo}}\right) \\ &= \text{SmoothL1}\left(\tilde{\mathbf{Q}}_{\text{gt}} - \hat{\mathbf{H}}\mathbf{q}_{\text{gt}}\right). \end{aligned} \tag{4}$$

Different from the regression loss, calculating the homography matrix $\hat{\mathbf{H}}$ will take all pairs of corresponding points into consideration. It is therefore a global loss for geometric constraint, which is used to guide the prediction of 3D positions from the ground truth 2D localization. On the other hand, by optimizing Eq. 4, $\hat{\mathbf{H}}$ is also enforced to be closer to the ground truth homography matrix. Another advantage of homography loss is that it is differentiable. It can be a plug-and-play module for any monocular 3D object detector, and servers as a strong spatial constraint for 3D localization of objects.

3.4. Case Study

As our novel homography loss can be plugged into any 3D object detector, we take the state-of-the-art detectors, ImVoxelNet [34] and MonoFlex [42], as examples, and illustrate how to seamlessly integrate our loss function into the network. As the main algorithm has been explained in Sec. 3.3, more details of the selection of predicted boxes and training strategies are presented here.

Anchor based method. ImVoxelNet [34] is a one-stage anchor-based monocular 3D detector, which transforms 2D image features into 3D space and regresses the positions of objects in BEV like LiDAR-based 3D detectors. Anchors

Table 1. 3D object detection performance of Car category on KITTI *test* set. The best scores are marked in bold (compared with monocular 3D detection methods that do not use extra data). 'Extra Data' lists the required extra information for each method, including **Depth** pretrained from a much larger dataset, **Temporal** information from multi frames, **LiDAR** with point cloud information and **Shape** of extra labelled 3D instance keypoints. All runtime values are collected from KITTI benchmark as well as the official paper and code.

Method	Extra Data	$AP_{3D\|R_{40}}$			$AP_{BEV\|R_{40}}$			Time(s)
		Easy	Moderate	Hard	Easy	Moderate	Hard	
Mono-PLiDAR [40]	Depth	10.76	7.50	6.10	21.27	13.92	11.25	0.10
PatchNet [22]	Depth	15.68	11.12	10.17	22.97	16.86	14.97	0.40
D4LCN [8]	Depth	16.65	11.72	9.51	22.51	16.03	12.55	0.20
MonoRUn [4]	Depth	19.65	12.30	10.58	27.94	17.34	15.24	0.07
Kinematic3D [2]	Temporal	19.07	12.72	9.17	26.69	17.52	13.10	0.12
DDMP-3D [37]	Depth	19.71	12.78	9.80	28.08	17.89	13.44	0.18
Aug3D-RPN [11]	Depth	17.82	12.99	9.78	26.00	17.89	14.18	0.08
DFR-Net [45]	Depth	19.40	13.63	10.35	28.17	19.17	14.84	0.18
CaDDN [32]	LiDAR	19.17	13.41	11.46	27.94	18.91	17.19	0.63
MonoEF [44]	Depth	21.29	13.87	11.71	29.03	19.70	17.26	0.03
Autoshape [20]	Shape	22.47	14.17	11.36	30.06	20.08	15.59	0.04
M3D-RPN [1]	-	14.76	9.71	7.42	21.02	13.67	10.23	0.16
SMOKE [19]	-	14.03	9.76	7.84	20.83	14.49	12.75	0.03
MonoPair [7]	-	13.04	9.99	8.65	19.28	14.83	12.89	0.06
RTM3D [17]	-	14.41	10.34	8.77	19.17	14.20	11.99	0.05
PGD-FCOS3D [38]	-	19.05	11.76	9.39	26.89	16.51	13.49	0.03
M3DSSD [21]	-	17.51	11.46	8.98	24.15	15.93	12.11	0.16
MonoDLE [24]	-	17.23	12.26	10.29	24.79	18.89	16.00	0.04
MonoRCNN [35]	-	18.36	12.65	10.03	25.48	18.11	14.10	0.07
ImVoxelNet [34]	-	17.15	10.97	9.15	25.19	16.37	13.58	0.20
ImVoxelNet(+homo)	-	20.10	12.99	10.50	29.18	19.25	16.21	0.20
MonoFlex [42]	-	19.94	13.89	12.07	28.23	19.75	16.89	0.03
MonoFlex(+homo)	-	**21.75**	**14.94**	**13.07**	**29.60**	**20.68**	**17.81**	**0.03**

with IoU > 0.6 will be considered as positives for training and each ground truth object will be assigned by several positive anchors that are served as potential proposals.

To calculate homography, we need to specify one-to-one matching point pairs for the predicted boxes and the ground truth boxes. Therefore, we choose the one with the highest classification score from positive proposals as a representative, which also keeps the consistency between classification and regression. As anchor-based detectors always produce stable proposals during training, we add the homography loss at the beginning of training and train the network from scratch. The loss function defined below consists of four parts, i.e., location loss L_{loc}, focal loss for classification L_{cls}, cross-entropy loss for direction L_{dir}, and additional homography loss L_{homo}:

$$L = \frac{1}{N_{pos}} (\lambda_{cls}L_{cls} + \lambda_{loc}L_{loc} + \lambda_{dir}L_{dir} + \lambda_{homo}L_{homo}), \quad (5)$$

where N_{pos} is the number of positive anchors, $\lambda_{cls} = 1.0, \lambda_{loc} = 2.0, \lambda_{dir} = 0.2, \lambda_{homo} = 0.2$. Note that, apart from L_{homo}, other loss terms and balancing weights are all adopted from [34].

Anchor-free based method. MonoFlex [42] is a one-stage monocular 3D detector based on CenterNet [43], which predicts projected 3D center, box (including depth, dimension, and orientation), and keypoints in different heads. As it is an anchor-free detector, the location of the representative box is automatically assigned as the 3D projected center in the heatmap head without selection. And the depth is regressed in the final head. The main difference is the training policy.

As 3D projected center and depth can define the coordinates in the image view and Bird's Eye View, these two components are the main contributors for homography loss. But the depth head is very unstable at the beginning of training, and the locations in the Bird's Eye View is also of low confidence, making the homography matrix distorted. Therefore, two strategies are proposed to solve the problem. Firstly, we make a delay by adding our homography loss after 40 epochs when the depth head is consistent and reliable. Secondly, we replicate the predicted boxes by using one of the components (3D projected center and depth), while replacing the other one with its ground truth values. Therefore, homography loss can be replicated three times and ensembled together. The main loss function can be described as a combination of classification loss for heatmap L_{hm}, regression loss for box size and rotation L_{box}, regression loss for keypoints of 3D boxes L_{kp}, and additional homography loss L_{homo}:

$$L = \frac{1}{N_{pos}}(\lambda_{hm}L_{hm} + \lambda_{box}L_{box} + \lambda_{kp}L_{kp} + \lambda_{homo}L_{homo}), \quad (6)$$

where N_{pos} is the number of positive predictions, $\lambda_{hm} = 1.0, \lambda_{box} = 1.0, \lambda_{kp} = 1.0, \lambda_{homo} = 0.2$.

Table 2. 3D object detection performance of Car category on KITTI *validation* set.

Method	$\mathrm{AP}_{3D|R_{40}}$			$\mathrm{AP}_{BEV|R_{40}}$		
	Easy	Moderate	Hard	Easy	Moderate	Hard
M3D-RPN [1]	14.53	11.07	8.65	20.85	15.62	11.88
MonoPair [7]	16.28	12.30	10.42	24.12	18.17	15.76
MonoRCNN [35]	16.61	13.19	10.65	25.29	19.22	15.30
MonoDLE [24]	17.45	13.66	11.68	24.97	19.33	17.01
ImVoxelNet(+homo)	21.44	14.88	12.08	29.85	21.17	17.77
MonoFlex(+homo)	**23.04**	**16.89**	**14.90**	**31.04**	**22.99**	**19.84**

Table 3. 3D object detection performance of Pedestrian and Cyclist on KITTI *test* set.

Method	Pedestrian $\mathrm{AP}_{3D|R_{40}}$			Cyclist $\mathrm{AP}_{3D|R_{40}}$		
	Easy	Moderate	Hard	Easy	Moderate	Hard
PGD-FCOS3D [38]	2.28	1.49	1.38	2.81	1.38	1.20
MonoEF [44]	4.27	2.79	2.21	1.80	0.92	0.71
D4LCN [8]	4.55	3.42	2.83	2.45	1.67	1.36
M3D-RPN [1]	4.92	3.48	2.94	0.94	0.65	0.47
DDMP-3D [37]	4.93	3.55	3.01	4.18	2.50	2.32
DFR-Net [45]	6.09	3.62	3.39	**5.69**	**3.58**	**3.10**
M3DSSD [21]	5.16	3.87	3.08	2.10	1.51	1.58
Aug3D-RPN [11]	6.01	4.71	3.87	4.36	2.43	2.55
MonoFlex [42]	9.43	6.31	5.26	4.17	2.35	2.04
MonoPair [7]	10.02	6.68	5.53	3.79	2.12	1.83
MonoRUn [4]	10.88	6.78	5.83	1.01	0.61	0.48
ImVoxelNet(+homo)	**12.47**	7.62	6.72	1.52	0.85	0.94
MonoFlex(+homo)	11.87	**7.66**	**6.82**	5.48	3.50	2.99

4. Experiments

4.1. Setup

Dataset and Evaluation Metrics. Our proposed method is evaluated on KITTI 3D Object Detection benchmark [10], which includes 7481 images for training and 7518 images for testing. The training set is split into 3712 samples for training and 3769 samples for validation as suggested in [6]. The classes are Car, Pedestrian, and Cyclist with three difficulty levels for each class, i.e., Easy, Moderate, and Hard. The official KITTI leaderboard is ranked on Moderate difficulty. Our method is evaluated on KITTI test set by submitting the detection results to the official server. For a fair comparison with other methods, we use official metrics, average precision (AP) with an IoU threshold of 0.7 for Car and 0.5 for both Pedestrian and Cyclist. In all experiments, the $\mathrm{AP}_{3D|R_{40}}$ results are reported for a comprehensive comparison with previous studies.

Implementation Details. We use the official implementations of ImVoxelNet [34] with ResNet50 [12] and MonoFlex [42] with DLA34 [41] as their backbones. We follow all the experimental settings of the original code and add our homography loss as an auxiliary loss. For ImVox-

elNet [34], we add the loss at the beginning and train 24 epochs. As for MonoFlex [42], the homography loss is added after 40 epochs and we train the network 80 epochs in total. We name these two new implementations as **ImVoxelNet(+homo)** and **MonoFlex(+homo)**, respectively.

4.2. Quantitative Results

Results of Car category on KITTI *test* set. As demonstrated in Tab. 1, the proposed method MonoFlex(+homo) achieves superior results on Car category compared with the previous methods, even including those with extra data, such as depth or LiDAR point clouds. To be specific, MonoFlex(+homo) achieves 1.81%, 1.05%, and 1.00% gains on the easy, moderate and hard settings, respectively. Besides, our ImVoxelNet(+homo) achieves 2.95%, 2.02%, and 1.45% gains over the original baseline, which shows its robustness and effectiveness.

Results of Car category on KITTI *validation* set. We also present our model's performance on the KITTI *validation* set in Tab. 2. Specifically, our method achieves the SOTA performance compared with the previous methods. Compared to MonoPair [7], our ImVoxelNet(+homo) and MonoFlex(+homo) get performance gain by 2.58%/4.59% for moderate setting at the 0.7 IoU threshold. This shows that our method is more capable of detecting hard examples in autonomous driving scenes by adding homography loss as an additional constraint.

Pedestrian/Cyclist detection on KITTI *test* set. For Pedestrian and Cyclist, we present the detection performance in Tab. 3. Our method MonoFlex(+homo) leads to the competitive performance in both categories. This shows our homography loss can also improve the performance for detecting small objects, e.g., human. MonoFlex(+homo) outperforms all other approaches in the Pedestrian category, with an 0.88% improvement from the previous best method (7.66% vs 6.78%). A possible reason is that human's standing point is a more reliable reference for computing the homography matrix.

4.3. Ablation Study

We conduct ablation studies to analyze the effects of our loss on Car category of the KITTI *validation* set. The default evaluation metric is $\mathrm{AP}_{3D|R_{40}}$.

4.3.1 Calculating Homography

To calculate the homography matrix, we use $\mathbf{q}_{gt}$ and $\tilde{\mathbf{Q}}_{pred}$ (Type 1) to construct the geometric constraints. Similarly, $\mathbf{q}_{pred}$ and $\tilde{\mathbf{Q}}_{gt}$ (Type 2) can also be selected. Therefore, we compare the performance of these two types in ImVoxelNet(+homo) and MonoFlex(+homo). The results are listed in Tab. 4 and 5. We can see that for those methods that predict in BEV domain like ImVoxelNet, Type 2 is more suit-

Table 4. Different settings of ImVoxelNet are evaluated on the *validation* set. By default, as shown in the top row, we combine homography type 2 (Sec. 4.3.1), representative proposal type 1 (Sec. 4.3.2), and weight of 0.2 (Sec. 4.3.4) with homography loss to obtain the best performance. Each row evaluates one specific setting compared with the default choice. The bottom row shows the comparison with projection loss (Sec. 5.1).

Homo		Proposal			Weight					Loss		$AP_{3D|R_{40}}$		
1	2	1	2	3	None	0.1	0.2	0.5	1.0	+homo	+proj	Easy	Moderate	Hard
	✔	✔					✔			✔		**21.44**	**14.88**	**12.08**
✓		✓					✓			✓		21.35	14.63	11.60
	✓		✓				✓			✓		19.41	14.21	11.63
	✓			✓			✓			✓		20.29	14.26	11.60
	✓	✓			✓					✓		20.20	13.85	11.41
	✓	✓				✓				✓		21.01	14.19	11.53
	✓	✓						✓		✓		20.43	14.13	11.48
	✓	✓							✓	✓		19.27	13.99	11.53
							✓				✓	20.51	14.13	11.49

able. As for those who predict in 2D images like MonoFlex, Type 1 gets higher performance. Therefore, the prediction domain can affect the final performance. So how to choose a proper type will finally depend on the specific application.

4.3.2 Representative Proposals

In anchor-based methods like ImVoxelNet, several anchors will be assigned to the same ground truth box based on IoU threshold. Therefore, we need to select the representative proposal from these positive proposals. Here, we have three strategies of selection: 1) the proposal with the highest classification score, 2) the proposal with the highest IoU score, 3) the average proposal of all positive anchors. We conduct the ablation experiment in Tab. 4. The result shows that the one with the highest classification score achieves the best performance at 14.88% of the moderate setting. It also shows our homography loss can strengthen the consistency between regression and classification heads.

4.3.3 Replicated Losses

For anchor-free methods, such as MonoFlex, the depth regression head can be very unstable at the beginning of training. To solve this problem, we refer to the replicated strategy in [27] and propose a replicated proposal strategy here to strengthen the robustness. The homography loss is replicated 3 times in total to get a reliable homography matrix. We conduct the ablation by four different settings: 1) $\mathbf{q}_{pred}$ + $\mathbf{Depth}_{pred}$ (the predicted depth), 2) $\mathbf{q}_{pred}$ + $\mathbf{Depth}_{gt}$ (the ground truth depth). 3) $\mathbf{q}_{gt}$ + $\mathbf{Depth}_{pred}$. 4) ensemble by adding the aforementioned three losses together. The results are shown in Tab. 5. We observe that the ensemble strategy has a better result due to sufficient constraints.

Table 5. Different settings of MonoFlex are evaluated on the *validation* set. By default, as shown in the top row, we combine homography type 1 (Sec. 4.3.1), ensembled losses (Sec. 4.3.3), and weight of 0.2 with homography loss to achieve the best result.

Homo		Replicated losses				Weight	Loss	$AP_{3D|R_{40}}$		
1	2	1	2	3	Ensemble	0.2	+homo	Easy	Moderate	Hard
✔					✔	✔	✔	**23.04**	**16.89**	**14.90**
	✓				✓	✓	✓	22.37	16.48	14.41
✓		✓				✓	✓	21.92	16.54	13.84
✓			✓			✓	✓	22.48	16.62	14.49
✓				✓		✓	✓	22.51	16.69	14.46

4.3.4 Loss Weight

To determine the final loss weight of ImVoxelNet(+homo) and MonoFlex(+homo), we also conduct experiments on loss weights. The results are shown in Tab. 4. We observe superior performance when the loss weight is 0.2. Therefore, we apply this configuration in training and get the final performance. For MonoFlex(+homo), we also do the same experiments and get 0.2 as a result. It shows our homography loss can be served as an auxiliary loss for detection.

4.4. Qualitative Results

From the qualitative results demonstrated in Fig. 5, with the proposed homography loss function, we can get superior performance for normal objects in the scene. Even for very challenging cases, such as small objects (distant Pedestrian and Car), and extremely truncated objects, our method can still successfully detect those well.

5. Discussions

5.1. Difference with Projection Loss

As shown in Eq. 2, the calibration parameters of the camera can be used to project a single predicted 3D keypoint onto a 2D image plane which will be further constrained by its corresponding 2D ground-truth value. It means that each training sample is considered individually, and the predicted 3D positions are also refined and optimized independently during network training. This is the key idea of the commonly used projection loss. However, for calculating the homography matrix, all pairs of correspondences will be involved in the computation, each pair of corresponding 2D/3D points will contribute two linear equations for solving Eq. 3. During backpropagation of the gradient of Eq. 4, $\hat{\mathbf{H}}$ is gradually optimized, that is to say, all the predicted $\tilde{\mathbf{Q}}_{pred}$ that are used for calculating the homography matrix will also be refined according to the chain rule. Therefore, homography loss can be leveraged to globally constrain the optimization of 3D localization. We compare projection loss with the proposed homography loss as shown in Tab. 4.

Figure 5. We visualize the results of 3D object detection using ImVoxelNet(+homo) on KITTI *val* set, where the orange represents the ground truth and our predicted results are colored in blue. The left column shows results of the network trained on the Car category only, and the right column is trained on three categories including Car, Pedestrian, and Cyclist. It is worth noting that, with the homography loss, it is possible to detect small targets and even truncated objects.

Table 6. Depth range statistics at the 0.7 and 0.5 IoU threshold.

Metric	ImVoxeNet	Depth Range (m)			
		0-10	10-20	20-30	>30
KITTI Moderate $AP_{3D|R_{40}}$	baseline@0.7	35.45	17.48	1.23	0.17
	+homo@0.7	**35.99**	**20.48**	**2.17**	**0.20**
	baseline@0.5	78.57	59.34	15.24	3.31
	+homo@0.5	**81.57**	**61.68**	**18.44**	**4.07**

5.2. Depth Range Statistics

In order to investigate why homography loss is useful for improving the accuracy of 3D detection. We design an experiment that divides the depth range into several segments as shown in Tab. 6 and gets the statistics for each interval. For fairness, the evaluation metric is also $AP_{3D|R_{40}}$ on the Car category of the KITTI *validation* set with the difficulty of moderate level. Obviously, we can see that in the area of 10 meters away, the effect of the detection algorithm with homography loss is much better than that of the baseline. Especially in the range of 10-20m, we obtain 3.0% and 2.34% gains over the baseline method with different IoU thresholds, respectively. This shows that our loss function is more effective for small target detection. The reason is that, as elaborated in Eq. 3, the ground truth 2D position q_{gt} on the image plane is used as guidance to correct the predicted 3D position $\tilde{Q}_{pred}$. The relative geometric relationship of objects on the image plane will be transferred to the corresponding 3D objects on BEV plane by homography loss. In other words, the improvement of the detection effect of distant objects in Tab. 6 is due to the homography

relationship, which refines the inaccurate estimated 3D positions to satisfy the overall geometric constraint.

5.3. Limitations

As stated in Sec. 3.3, we assume that the ground plane is flat and use the simplified 2D coordinates $\tilde{Q} = [x, y]^T$ on BEV plane to replace the original 3D points $Q = [x, y, z]^T$. However, in practice, as pointed out in [44], usually the road is not smooth and has slight fluctuation, it will influence the accuracy of 3D detection.

6. Conclusion

In this paper, we propose a differentiable loss function, named as *homography loss*, which is a plug-and-play module that can be integrated into any monocular 3D detector, to help globally optimize the 3D positions of all the objects, instead of taking each object as an independent sample during training. Homography loss also fully exploits the inherent connection between 2D image space and 3D Bird's Eye View and constrains the optimization of 3D positions under the guidance of 2D localization, which is demonstrated to be useful for detecting small targets or highly truncated objects. In the future work, we will consider how to avoid the assumption of flatness of the ground.

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 (ICCV)*, 2019. 2, 5, 6

[2] Garrick Brazil, Gerard Pons-Moll, Xiaoming Liu, and Bernt Schiele. Kinematic 3d object detection in monocular video. In *European Conference on Computer Vision (ECCV)*, 2020. 5

[3] 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 (CVPR)*, 2017. 2

[4] Hanshen Chen, Yuyao Huang, Wei Tian, Zhong Gao, and Lu Xiong. Monorun: Monocular 3d object detection by reconstruction and uncertainty propagation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 5, 6

[5] Xiaozhi Chen, Kaustav Kundu, Ziyu Zhang, Huimin Ma, Sanja Fidler, and Raquel Urtasun. Monocular 3d object detection for autonomous driving. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2

[6] 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. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems (NIPS)*, 2015. 6

[7] 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 (CVPR)*, 2020. 1, 3, 5, 6

[8] Mingyu Ding, Yuqi Huo, Hongwei Yi, Zhe Wang, Jianping Shi, Zhiwu Lu, and Ping Luo. Learning depth-guided convolutions for monocular 3d object detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2020. 5, 6

[9] Jiaojiao Fang, Lingtao Zhou, and Guizhong Liu. 3d bounding box estimation for autonomous vehicles by cascaded geometric constraints and depurated 2d detections using 3d results. *arXiv preprint arXiv:1909.01867*, 2019. 2

[10] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012. 6

[11] Chenhang He, Jianqiang Huang, Xian-Sheng Hua, and Lei Zhang. Aug3d-rpn: Improving monocular 3d object detection by synthetic images with virtual depth. *arXiv preprint arXiv:2107.13269*, 2021. 5, 6

[12] Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 6

[13] Tong He and Stefano Soatto. Mono3d++: Monocular 3d vehicle detection with two-scale 3d hypotheses and task priors. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2019. 2

[14] Abhijit Kundu, Yin Li, and James M Rehg. 3d-rcnn: Instance-level 3d object reconstruction via render-and-compare. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2018. 2

[15] 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 (CVPR)*, 2019. 1, 2, 3

[16] Peixuan Li and Huaici Zhao. Monocular 3d detection with geometric constraint embedding and semi-supervised training. *IEEE Robotics and Automation Letters*, 6(3):5565–5572, 2021. 3

[17] Peixuan Li, Huaici Zhao, Pengfei Liu, and Feidao Cao. Rtm3d: Real-time monocular 3d detection from object keypoints for autonomous driving. In *European Conference on Computer Vision (ECCV)*, 2020. 1, 2, 3, 5

[18] Lijie Liu, Jiwen Lu, Chunjing Xu, Qi Tian, and Jie Zhou. Deep fitting degree scoring network for monocular 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[19] Zechen Liu, Zizhang Wu, and Roland Tóth. Smoke: Single-stage monocular 3d object detection via keypoint estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2020. 2, 5

[20] Zongdai Liu, Dingfu Zhou, Feixiang Lu, Jin Fang, and Liangjun Zhang. Autoshape: Real-time shape-aware monocular 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 5

[21] Shujie Luo, Hang Dai, Ling Shao, and Yong Ding. M3dssd: Monocular 3d single stage object detector. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 5, 6

[22] Xinzhu Ma, Shinan Liu, Zhiyi Xia, Hongwen Zhang, Xingyu Zeng, and Wanli Ouyang. Rethinking pseudo-lidar representation. In *European Conference on Computer Vision (ECCV)*, 2020. 5

[23] Xinzhu Ma, Zhihui Wang, Haojie Li, Pengbo Zhang, Wanli Ouyang, and Xin Fan. Accurate monocular 3d object detection via color-embedded 3d reconstruction for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 2

[24] Xinzhu Ma, Yinmin Zhang, Dan Xu, Dongzhan Zhou, Shuai Yi, Haojie Li, and Wanli Ouyang. Delving into localization errors for monocular 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 5, 6

[25] 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 (CVPR)*, 2017. 1, 2, 3

[26] Andretti Naiden, Vlad Paunescu, Gyeongmo Kim, Byeong-Moon Jeon, and Marius Leordeanu. Shift r-cnn: Deep monocular 3d object detection with closed-form geometric constraints. In *IEEE International Conference on Image Processing (ICIP)*, 2019. 1, 2, 3

[27] Dennis Park, Rares Ambrus, Vitor Guizilini, Jie Li, and Adrien Gaidon. Is pseudo-lidar needed for monocular 3d object detection? In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 2, 7

[28] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems (NIPS)*, 2019. 4

[29] Liang Peng, Fei Liu, Zhengxu Yu, Senbo Yan, Dan Deng, and Deng Cai. Lidar point cloud guided monocular 3d object detection. *arXiv preprint arXiv:2104.09035*, 2021. 1

[30] 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 (CVPR)*, 2020. 2

[31] Zengyi Qin, Jinglu Wang, and Yan Lu. Monogrnet: A geometric reasoning network for monocular 3d object localization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2019. 2

[32] Cody Reading, Ali Harakeh, Julia Chae, and Steven L Waslander. Categorical depth distribution network for monocular 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2, 5

[33] Thomas Roddick, Alex Kendall, and Roberto Cipolla. Orthographic feature transform for monocular 3d object detection. *arXiv preprint arXiv:1811.08188*, 2018. 2

[34] Danila Rukhovich, Anna Vorontsova, and Anton Konushin. Imvoxelnet: Image to voxels projection for monocular and multi-view general-purpose 3d object detection. *arXiv preprint arXiv:2106.01178*, 2021. 2, 3, 4, 5, 6

[35] Xuepeng Shi, Qi Ye, Xiaozhi Chen, Chuangrong Chen, Zhixiang Chen, and Tae-Kyun Kim. Geometry-based distance decomposition for monocular 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 5, 6

[36] Andrea Simonelli, Samuel Rota Bulo, Lorenzo Porzi, Manuel López-Antequera, and Peter Kontschieder. Disentangling monocular 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 2

[37] Li Wang, Liang Du, Xiaoqing Ye, Yanwei Fu, Guodong Guo, Xiangyang Xue, Jianfeng Feng, and Li Zhang. Depth-conditioned dynamic message propagation for monocular 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 5, 6

[38] Tai Wang, Xinge ZHU, Jiangmiao Pang, and Dahua Lin. Probabilistic and geometric depth: Detecting objects in perspective. In *5th Annual Conference on Robot Learning*, 2021. 5, 6

[39] 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 (CVPR)*, 2019. 2

[40] Xinshuo Weng and Kris Kitani. Monocular 3d object detection with pseudo-lidar point cloud. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, 2019. 2, 5

[41] Fisher Yu, Dequan Wang, and Trevor Darrell. Deep layer aggregation. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 6

[42] Yunpeng Zhang, Jiwen Lu, and Jie Zhou. Objects are different: Flexible monocular 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2, 4, 5, 6

[43] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. In *arXiv preprint arXiv:1904.07850*, 2019. 5

[44] Yunsong Zhou, Yuan He, Hongzi Zhu, Cheng Wang, Hongyang Li, and Qinhong Jiang. Monocular 3d object detection: An extrinsic parameter free approach. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 5, 6, 8

[45] Zhikang Zou, Xiaoqing Ye, Liang Du, Xianhui Cheng, Xiao Tan, Li Zhang, Jianfeng Feng, Xiangyang Xue, and Errui Ding. The devil is in the task: Exploiting reciprocal appearance-localization features for monocular 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 5, 6