NeurOCS: Neural NOCS Supervision for Monocular 3D Object Localization

Zhixiang Min[1] Bingbing Zhuang[2] Samuel Schulter[2] Buyu Liu[2]
Enrique Dunn[1] Manmohan Chandraker[2]
[1]Stevens Institute of Technology [2]NEC Laboratories America

Abstract

Monocular 3D object localization in driving scenes is a crucial task, but challenging due to its ill-posed nature. Estimating 3D coordinates for each pixel on the object surface holds great potential as it provides dense 2D-3D geometric constraints for the underlying PnP problem. However, high-quality ground truth supervision is not available in driving scenes due to sparsity and various artifacts of Lidar data, as well as the practical infeasibility of collecting per-instance CAD models. In this work, we present NeurOCS, a framework that uses instance masks and 3D boxes as input to learn 3D object shapes by means of differentiable rendering, which further serves as supervision for learning dense object coordinates. Our approach rests on insights in learning a category-level shape prior directly from real driving scenes, while properly handling single-view ambiguities. Furthermore, we study and make critical design choices to learn object coordinates more effectively from an object-centric view. Altogether, our framework leads to new state-of-the-art in monocular 3D localization that ranks 1st on the KITTI-Object [16] benchmark among published monocular methods.

1. Introduction

Localization of surrounding vehicles in 3D space is an important problem in autonomous driving. While Lidar [32, 59, 72] and stereo [32, 59, 72] methods have achieved strong performances, monocular 3D localization remains a challenge despite recent progress in the field [4, 35, 43].

Monocular 3D object localization can be viewed as a form of 3D reconstruction, with the goal to estimate the 3D extent of an object from single images. While single-view 3D reconstruction is challenging due to its ill-posed nature, learned priors combined with differentiable rendering [65] have recently emerged as a powerful technique, which has the potential to improve 3D localization as well. Indeed, researchers [2, 31, 78, 79] have applied it as a means for object pose optimization in 3D localization. However, difficulties remain due to pose optimization being ambiguous under challenging photometric conditions (such as textureless vehicle surfaces) and geometric occlusions ubiquitous in driving scenes. The question of how differentiable rendering may be best explored in 3D localization remains under-studied. Our work proposes a framework to unleash its potential – instead of using differentiable rendering for pose optimization, we use it with annotated ground truth pose to provide high-quality supervision for image-based shape learning, leading to a new state-of-the-art in 3D localization performance.

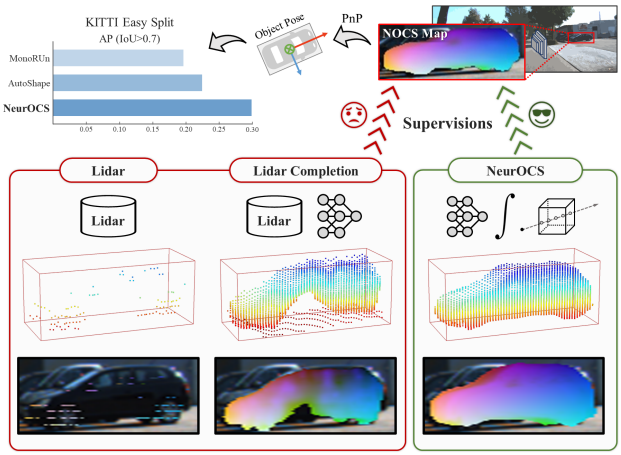


Figure 1. **NeurOCS** learns category-level shape model in real driving scenes to provide dense and clean NOCS supervision through differentiable rendering, leading to new state-of-the-art 3D object localization performance.

Our framework relies on the machinery of Perspective-n-Point (PnP) [33] pose estimation, which uses 2D-3D constraints to explicitly leverage geometric principles that lend itself well to generalization. In particular, learning 3D object coordinates for every visible pixel on the object surface, known as normalized object coordinate space (NOCS) [66], provides a dense set of constraints. However, despite NOCS-based pose estimation dominating indoor benchmarks [21], their use in real driving scenes has been limited primarily due to lack of supervision – it is nontrivial to obtain accurate per-instance reconstructions or CAD models in road scenes. Lidar or its dense completion [23] are natural alternatives

as pseudo ground truth [7], but they are increasingly sparse or noisy on distant objects with reflective surfaces. Synthetic data is a potential source of supervision [78, 79], but is inherently restricted by domain gap to real scenes.

In this work, we propose *NeurOCS* that leverages neural rendering to obtain effective NOCS supervision. Firstly, we propose to learn category-level shape reconstruction from real driving scenes with object masks and 3D object boxes using Neural Radiance Field (NeRF) [44]. The shape reconstruction is then rendered into NOCS maps that serve as the pseudo ground-truth for a network dedicated to regress NOCS from images. Specifically, NeurOCS learns category-level shape representation as a latent grid [50, 51] with low-rank structure, consisting of a canonical latent grid plus several deformation bases to account for instance variations. With single-view ambiguities handled by a KL regularization [26] and dense shape prior, we show that the NOCS supervision so obtained yields strong 3D localization performance, even when the shape model is trained without using Lidar data or any CAD models. Our NOCS supervision is illustrated in Fig. 1 in comparison with Lidar and its dense completion. We also note that NeRF rendering is only required during training, without adding computational overhead to inference. We show NeurOCS is complementary to direct 3D box regression [43, 55], and their fusion further boosts the performance.

Further, we study crucial design choices in image-conditioned NOCS regression. For example, as opposed to learning NOCS in a scene-centric manner with the full image as network input, we learn in an object-centric view by cropping objects without scene context, which is demonstrated to especially benefit the localization of distant or occluded objects. Our extensive experiments study key choices that enable NeurOCS to achieve top-ranked accuracy for monocular 3D localization on the KITTI-Object benchmark [16].

In summary, our contributions include:

- We propose a framework to obtain neural NOCS supervision through differentiable rendering of the category-level shape representation learned in real driving scenes.
- We drive the learning with deformable shape representation as latent grids with careful regularizations, as well as effective NOCS learning from an object-centric view.
- Our insights translate to state-of-the-art performance which achieves a top rank in KITTI benchmark [16].

2. Related Work

Direct regression methods. These methods directly regress 3D bounding boxes parameters [3, 9, 48, 62, 74, 81]. In light of depth being the most critical factor [25, 43], many works develop *depth*-guided localization, including depth-aware networks [3, 13, 24, 28], depth-conditioned message propagation [67], depth-equivariant network [28], depth-guided feature projection [58], depth-based feature enhancement [1], depth pretraining for knowledge distillation [53], depth from motion [69], and object depth decomposition [53] for affine data augmentation. In addition, [9, 17, 39, 45, 47, 57, 82, 83] utilize the ground plane as depth prior for localization. Researchers [42, 56, 61, 68, 71, 75] also convert depth maps into the pseudo-Lidar representation to directly apply advanced Lidar-based methods. [22, 54, 73] utilize pseudo labels or teacher-student training to learn from unlabeled data.

Geometric reasoning methods. These methods [27, 48] solve object poses with 2D-3D perspective constraints, such as box edge correspondences [48], object heights [28, 41, 60], sparse keypoints [6, 34, 40], or edges [35] on the object surface. EPro-PnP [8] learns localization-friendly correspondences without explicit shape constraints, but it has limited performance in KITTI that has relatively small amount of training data. Notably, a line of methods learn per-pixel object coordinates known as NOCS [66] to establish dense constraints. In view of the noisy nature of Lidar, MonoRUn [7] proposes a self-supervised way to learn NOCS by minimizing reprojection error, which nonetheless does not provide strong shape constraints and still requires Lidar to achieve good accuracy. Some methods [78, 79] resort to synthetic data for NOCS supervision, which suffers from domain gap. In this work, we show that NOCS can be learned effectively from high-quality NeRF-rendered supervision. This also draws connection to AutoShape [40] as an auto-labeling pipeline that uses CAD models to generate pseudo ground truth for sparse keypoints. Our method distinguishes itself by directly learning dense NOCS from real driving scenes and working well even without using CAD models and Lidar supervision. Also worth noting is MonoJSG [37] that leverages learned NOCS to perform cost volume based object depth refinement, further validating the merits of NOCS.

Hybrid methods. Regression-based and geometry-based methods are not mutually exclusive but rather complementary. [28, 41, 80] combine depth from regression and depth from height for improved accuracy. MonoDDE [36] further enriches the set of depth cues and fuses them in an end-to-end trainable framework. While focusing on NOCS-based geometric reasoning, our method when fused with direct depth regression results in improvements as well (Sec. 4.2), demonstrating their desirable complementary nature.

Differentiable rendering. A number of works [2, 31, 46, 78, 79] have applied differentiable rendering to the 3D localization problem. They typically use it as a means to optimize object pose by minimizing photometric or feature-metric error through a render-and-compare manner. However, pose optimization may be highly ambiguous under the challenging conditions (Sec. 1) in real driving scenes, especially with a single view. Hence these methods do not currently dominate the standard 3D localization benchmark [16]. In

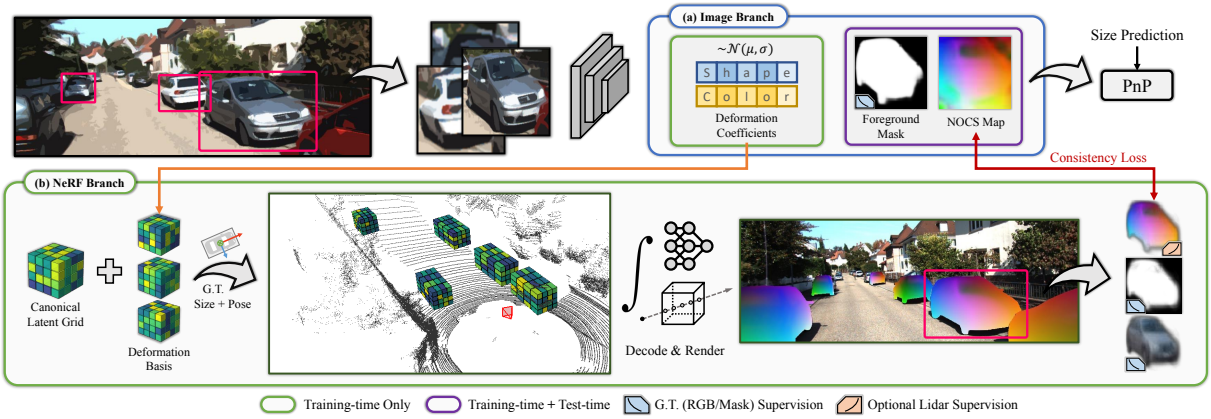


Figure 2. **Overview of NeurOCS.** Given each detected object, our network predicts the object mask, NOCS map, and its deformation coefficients associated with a categorical NeRF model. The jointly trained NeRF renders the NOCS supervision for the network prediction branch. During inference, only the predicted object mask and NOCS map are consumed by PnP for localization.

this work, we use differentiable rendering as well but with annotated object poses, and obtain NOCS supervision that leads to state-of-the-art performance in 3D localization. We also note that the recent success of NeRF [44] continues to promote differentiable rendering research [30, 49, 52] in driving scenes. However, they assume known 3D object boxes as input, with a focus solely on rendering.

3. Method

3.1. Overview

Problem Formulation. We aim to localize each object as an enclosing 3D box parameterized by its 3D dimension $\mathbf{s} = [l, h, w]$ and the 4-DoF pose parameters including the yaw-angle θ and object center $\mathbf{t} = [x, y, z]$. Our work studies the NOCS-based approach and solves the pose as a PnP problem. Specifically, we denote the position of each 3D point on the object surface as $\mathbf{x}_i = [x_i^{[x]}, x_i^{[y]}, x_i^{[z]}]^T$ represented in its object coordinate system. $\mathbf{x}_i$ is further decoupled as the scale-invariant coordinate $\mathbf{o}_i$ in normalized object coordinate space (NOCS) times the object size $\mathbf{s}_i$, i.e. $\mathbf{x}_i = \mathbf{o}_i \odot \mathbf{s}_i$, for effective learning. With network-predicted $\mathbf{o}_i$ and $\mathbf{s}_i$ on a pixel with normalized camera coordinate $\mathbf{p}_i = [u_i, v_i, 1]^T$, we solve the pose by minimizing the reprojection error as

$$\underset{\mathbf{R}_\theta, \mathbf{t}}{argmin} \sum_i \rho\Big(w_i\big(\frac{\mathbf{R}_\theta \mathbf{x}_i + \mathbf{t}}{[\mathbf{R}_\theta \mathbf{x}_i + \mathbf{t}]_z} - \mathbf{p}_i\big)\Big), \qquad (1)$$

where $[\cdot]_z$ denotes the z-axis component, $\mathbf{R}_\theta$ is the rotation matrix form of the yaw θ, w_i is the confidence weight for each prediction and ρ denotes a robust M-estimator, where we use Huber loss throughout this work.

Framework Overview. Fig. 2 shows an overview of NeurOCS's key components. NeurOCS predicts NOCS as input to PnP estimation of object pose, with a NeRF used to render pseudo-ground truth NOCS for supervision. Specifically, we first use a separately trained base 3D detector [43, 55] to obtain 2D detections with their 3D size predictions. We then crop each detected object and use a ResNet50 [19] followed by a few regression heads to predict its object mask, NOCS map, as well as two sets of coefficients representing the shape and color of the object instance. The coefficients account for instance variations and are used for deforming a NeRF-based shape model, represented by latent grids as detailed in Sec. 3.2 and regularized with deformation and shape priors discussed in Sec. 3.3. We jointly train the NeRF and the prediction network – the NeRF is trained with the ground-truth *3D boxes* (i.e. pose and size), *object mask*, *RGB color*, and optionally *Lidar* data; and the prediction network is trained with the ground-truth *object mask* and the NOCS map rendered from NeRF. During inference, NeRF is switched off and only the predicted NOCS map and object mask are consumed by PnP for pose estimation (Sec. 3.4), where we also predict a confidence score. The PnP solution may be fused with the complementary direct depth prediction additionally regressed by the base 3D object detector. Note that our prediction network is decoupled from the base 3D detector, for reasons detailed in Sec. 3.5.

3.2. Categorical Shape Model

Shape Representation. Inspired by InstantNeRF [51], we employ a 3D latent grid ϕ as our shape representation owing to its efficiency in training and inference. In particular, we use a multi-resolution dense grid as in [50] to model shape in a unit cube. For an input NOCS point $\mathbf{o}_i$, we query the grid by trilinear interpolation and stack the multi-resolution outputs to return a D dimensional feature vector, which can be efficiently decoded by a small MLP network to output density and RGB color. Our shape representation does not model the view-dependent effects as in vanilla NeRF.

Deformation. To model the categorical shape variations, we use a set of learnable 3D latent grids to compose a

low-rank deformable shape representation. We define a canonical latent grid ϕ_μ and a set of deformation grid basis $\{\phi_i \mid i = 1 \cdots B\}$ where B is the number of bases. For each object, given a coefficient $\mathbf{z} \in \mathbb{R}^B$ predicted by network, we construct an instance-specific latent grid Φ by deforming ϕ_μ with the linearly combined bases:

$$\Phi = \phi_\mu + \frac{\sum_{i=1}^{B} z_i \phi_i}{B}. \tag{2}$$

We maintain separate latent grids and coefficients for shape and color. By sharing the small number ($<<$ grid dimension) of bases, the latent grid is in a low-rank space that forces the deformation to explore categorical commonalities in shape.

Object Volume Rendering. For each object, we use the ground-truth 3D box (i.e. size and pose) to transform the viewing ray into its normalized object coordinate, denoted as $\mathbf{r}^{[\gamma]} = (\mathbf{q} + \gamma \mathbf{d}) \circ \frac{1}{\mathbf{s}}$, where $\mathbf{q}$, $\mathbf{d}$ and γ respectively denote the camera center, viewing direction and distance along the ray. The color $\mathbf{c}(\mathbf{r})$ can be rendered following [44] as

$$\begin{aligned} \mathbf{c}(\mathbf{r}) &= \int_{\gamma_n}^{\gamma_f} \alpha(\mathbf{r}^{[\gamma]})\, \Phi^{[\sigma]}\big(\mathbf{r}^{[\gamma]}\big)\, \Phi^{[\mathbf{c}]}\big(\mathbf{r}^{[\gamma]}\big)\, d\gamma, \\ \textit{and} \quad \alpha(\mathbf{r}^{[\gamma]}) &= exp\Big(-\int_{\gamma_n}^{\gamma} \Phi^{[\sigma]}\big(\mathbf{r}^{[\zeta]}\big) d\zeta\Big), \end{aligned} \tag{3}$$

where $\Phi^{[\mathbf{c}]}(.) \in \mathbb{R}^3$ and $\Phi^{[\sigma]}(.) \in \mathbb{R}^1$ denote decoding the latent grid Φ at a given query point into color and density, respectively. The near and far distance γ_n and γ_f are given by the ray intersection with the 3D box; therein points are sampled as illustrated in Fig. 2. $\alpha(\mathbf{r}^{[\gamma]})$ is the accumulated transmittance along the ray. The occupancy map indicating object mask is rendered by

$$\mathbf{m}(\mathbf{r}) = \int_{\gamma_n}^{\gamma_f} \alpha(\mathbf{r}^{[\gamma]})\, \Phi^{[\sigma]}\big(\mathbf{r}^{[\gamma]}\big)\, d\gamma. \tag{4}$$

We render the NOCS map by integrating the NOCS as

$$\mathbf{o}(\mathbf{r}) = \frac{\int_{\gamma_n}^{\gamma_f} \alpha(\mathbf{r}^{[\gamma]})\, \Phi^{[\sigma]}\big(\mathbf{r}^{[\gamma]}\big)\, \mathbf{r}^{[\gamma]}\, d\gamma}{\mathbf{m}(\mathbf{r})}. \tag{5}$$

Shape Losses. Our shape model is trained by several L_2 losses including occupancy loss, color (RGB) loss and additional NOCS supervision from lidar and its completion [23],

$$\mathcal{L}_{shape} = \mathcal{L}_{occ} + \mathcal{L}_{rgb} + \underbrace{\big(\mathcal{L}_{lidar} + \mathcal{L}_{licomp}\big)}_{optional}. \tag{6}$$

The occupancy loss is supervised by the ground-truth mask containing 3 categories including foreground, background and unknown (usually due to occlusion), similarly to [49]. We enforce the occupancy to be 1 at foreground, 0 at background and skip the unknowns (see supplementary for examples). $\mathcal{L}_{rgb}$, $\mathcal{L}_{lidar}$, and $\mathcal{L}_{licomp}$ are only applied at foreground regions. For $\mathcal{L}_{lidar}$ and $\mathcal{L}_{licomp}$, we convert point clouds inside the 3D box into NOCS and supervise the corresponding pixels. While being an indirect shape regularization, we found the photometric constraints from $\mathcal{L}_{rgb}$ benefit the performance. For training the latent bases, we only select high-quality examples that are at least k pixels in height ($k = 40$) and have no occlusions. Otherwise, we freeze the latent bases and only optimize their coefficient predictions.

3.3. Categorical Shape Regularization

Deformation Regularization. Despite the shared low-rank deformation bases across objects, their deformations may still not be well-regularized under an ill-posed single-view reconstruction. Hence, we predict $\mathbf{z}$ as a learnable Gaussian distribution $q(\mathbf{z} \mid \mathbf{I})$ and add the KL divergence loss [26] as a regularization minimizing information gain as in VAEs [26],

$$\mathcal{L}_{kl} = KL\big(q(\mathbf{z} \mid \mathbf{I}) \mid\mid p(\mathbf{z})\big), \tag{7}$$

where $p(\mathbf{z}) \sim \mathcal{N}(0, 1)$ is the prior latent distribution, $\mathbf{I}$ indicates the object image. $\mathbf{z}$ is sampled from $q(\mathbf{z} \mid \mathbf{I})$ using the reprameterization trick as in VAEs [26] for optimization. This loss prevents redundant deformations and yields cleaner shapes as shown in supplementary.

Dense Shape Prior. In the absence of Lidar supervision, the occupancy loss as a major shape cue is conceptually a shape-from-silhouette reconstruction, which suffers from the familiar visual hull ambiguity [12] (see supp. for illustration). Inspired by [76], we apply a dense shape prior to favor solid over empty space if both are possible solutions,

$$\mathcal{L}_{dense} = \frac{\sum_{s}^{S} exp\big(-\Phi^{[\sigma]}(\mathbf{o}_s) \cdot d\big)}{S}, \tag{8}$$

where the prior is applied to S randomly sampled NOCS $\mathbf{o}_s$. $d = 0.05$ is a hyper-parameter indicating an interval. Note we only apply this prior in the absence of Lidar supervision.

3.4. 3D Object Localization

NOCS Prediction. For learning NOCS, we enforce the consistency between the network-predicted NOCS and the NeRF-rendered ones from Eq.(5), yielding the loss

$$\mathcal{L}_{nocs} = \frac{\sum_{i \in \Omega_{fg}} ||\mathbf{o}_i^{[pred]} - \mathbf{o}_i^{[render]}||^2}{\big|\Omega_{fg}\big|}, \tag{9}$$

where Ω_{fg} denotes all pixels within the ground-truth object mask. Finally, we regress a foreground object mask using a simple L_2 loss with the ground truth.

PnP Solver and Score Prediction. Combining the NOCS with the object size prediction from the base detector, we solve the PnP problem (Eq. (1)) using Levenberg–Marquardt with an outlier-robust initialization scheme same as [8]. We use the predicted foreground probability along with a learned uncertainty map (detailed in supp.) as the per-pixel weight.

This step yields the estimated 3D box. A complete 3D detector also needs to return confidence scores indicating the accuracy. So, we add a regression head with L_2 loss to predict the IoU between the estimated and ground truth 3D box similarly to [7]. This head takes as input the object feature map, the predicted NOCS map, and a Jacobian map. The Jacobian map is obtained by the partial derivatives of the PnP loss (Eq. (1)) w.r.t. object center $\mathbf{t}$ at each pixel, evaluated at the solved pose. Specifically,

$$\frac{\partial \mathbf{r}_i^{[rep]}}{\partial \mathbf{t}} = \begin{bmatrix} \frac{\partial [\mathbf{r}_i^{[rep]}]_x}{\partial [\mathbf{t}]_x} & 0 & \frac{\partial [\mathbf{r}_i^{[rep]}]_x}{\partial [\mathbf{t}]_z} \\ 0 & \frac{\partial [\mathbf{r}_i^{[rep]}]_y}{\partial [\mathbf{t}]_y} & \frac{\partial [\mathbf{r}_i^{[rep]}]_y}{\partial [\mathbf{t}]_z} \\ 0 & 0 & 0 \end{bmatrix}, \tag{10}$$

where we flatten the non-zero elements as a feature vector for each pixel. The Jacobian measures the stability and correlates with the uncertainty [14] of the solved pose, thus supplies the network with informative signals to reason accuracy. Both the NOCS and Jacobian map are detached here, with gradients backpropagated through the object feature map only. Finally, we multiply the predicted IoU score with the confidence score from the base detector as the final score.

Scale Fusion. The metric scale of monocular localization is inherently ambiguous. In particular, the scale in PnP-based methods is solely determined by the object size prediction and could be unreliable [36]. Here, we propose a simple yet effective method that fuses into our PnP method the direct object depth d_{pred} additionally regressed by our base 3D detector. We first update object size prediction $\mathbf{s}$ as

$$\mathbf{s}' = \frac{d_{pred} + [\mathbf{t}]_z}{2 \cdot [\mathbf{t}]_z} \mathbf{s}. \tag{11}$$

$\mathbf{s}'$ inherently averages $\mathbf{s}$ with an optimal size $\frac{d_{pred}}{[\mathbf{t}]_z}\mathbf{s}$ that yields the prediction depth d_{pred} in the current PnP problem. Next, we scale the translation estimation $\mathbf{t}$ in tandem

$$\mathbf{t}' = \frac{d_{pred} + [\mathbf{t}]_z}{2 \cdot [\mathbf{t}]_z} \mathbf{t}. \tag{12}$$

This retains the optimality of our pose with the reprojection error intact, and yet the scale is fused across object size prediction and object depth prediction for robustness. We have also attempted more sophisticated fusion as discussed in supplementary, but found this scheme to be sufficient.

3.5. Key Design Choices

Visual Scope Matters. We discuss a critical design choice among two common strategies in obtaining object-level feature for NOCS regression, as illustrated in Fig. 3. One option (*e.g.* [79]) is to simply crop each object from the input image and regress NOCS from an *object-centric* view. This is opposed to another *scene-centric* strategy that takes the full image as network input and then crops the region-of-interest (RoI) from deep feature maps (*e.g.* from a detection backbone) to regress NOCS (e.g. [7, 37]). The impact of this distinction on the NOCS learning and the subsequent 3D localization has not been well-understood. We conduct extensive experiments in Sec. 4.3.2 to study this and observe the object-centric strategy to be significantly superior.

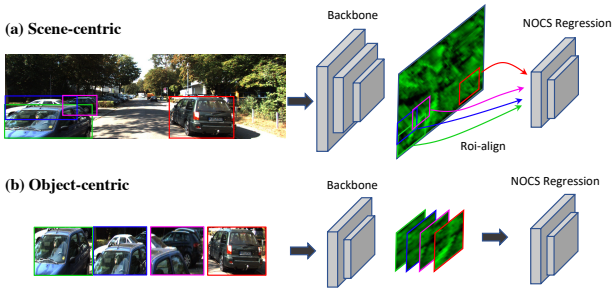


Figure 3. Illustration of the scene-centric and object-centric training scheme that are shown to be an important design choice.

Intuitively, we note that the scene-centric strategy retains rich context which is certainly essential for context-dependent tasks such as object depth estimation. However, object appearance alone has sufficient information to learn NOCS, as it is an intrinsic 3D property of the object. We conjecture that an object-centric view without scene context may enforce the network to learn NOCS the hard way – a strict mapping from object appearance to NOCS. Conversely, rich context in scene-centric view may cause context bias [63], allowing the network to rely on context instead of object appearance, which may hamper generalization with larger input variations. In Sec. 4.3.2, we observe that such an object-centric reasoning yields greater benefits for the challenging cases of distant or occluded objects.

Implementation Details. We combine all losses for joint training, with the weight of each term in supplementary. We apply a simple test-time augmentation by averaging inference on the flipped image. We use the code of [55] as base 3D detector. Detailed network architecture, computation efficiency, and more technical details are in supplementary.

4. Experiments

4.1. Dataset and Evaluation Metrics

KITTI. Following existing works we use KITTI-Object [16] dataset as the main evaluation benchmark, focusing on the *Car* category. This dataset contains a total of 7481/7518 training/test images. For ablation purpose the former is further split into 3716/3769 training and validation images [10]. Our framework uses the instance mask annotations provided by [20]. We use the standard evaluation metric AP_{40} [62] - the average precision sampled at 40 recall positions in the precision-recall curve. The AP is computed for both

Class	Method	Venue	Cat.	3D AP_{40} - Test			3D AP_{40} - Val			BEV AP_{40} - Test			BEV AP_{40} - Val		
				Easy	Moderate	Hard	Easy	Moderate	Hard	Easy	Moderate	Hard	Easy	Moderate	Hard
Regression	Monodle [43]	CVPR21	E	17.23	12.26	10.29	17.45	13.66	11.68	24.79	18.89	16.00	24.97	19.33	17.01
	MonoEF [82]	CVPR21	E	21.29	13.87	11.71	-	-	-	29.03	19.70	17.26	-	-	-
	DDMP-3D [67]	CVPR21	E	19.71	12.78	9.80	28.12	20.39	16.34	28.08	17.89	13.44	-	-	-
	CaDDN [58]	CVPR21	E	19.17	13.41	11.46	23.57	16.31	13.84	27.94	18.91	17.19	-	-	-
	GrooMeD-NMS [29]	CVPR21	E	18.10	12.32	9.65	19.67	14.10	10.47	26.19	18.27	14.05	27.38	19.75	15.92
	PCT [68]	NeurIPS21	E	21.00	13.37	11.31	38.39	27.12	24.11	29.65	19.03	15.92	47.16	34.65	28.47
	MonoGround [57]	CVPR22	E	21.37	14.36	12.62	25.24	18.69	15.58	30.07	20.47	17.74	32.68	24.79	20.56
	HomoLoss [17]	CVPR22	E	21.75	14.94	13.07	23.04	16.89	14.90	29.60	20.68	17.81	31.04	22.99	19.84
	MonoDTR [24]	CVPR22	E	21.99	15.39	12.73	24.52	18.57	15.51	28.59	20.38	17.14	33.33	25.35	21.68
	PseudoStereo [11]	CVPR22	E	23.74	17.74	15.14	35.18	24.15	20.35	32.84	23.67	20.64	-	-	-
	DID-M3D [55]	ECCV22	E	24.40	16.29	13.75	25.38	17.07	14.06	32.95	22.76	19.83	33.91	24.00	19.52
Geometric	MonoRUn [7]	CVPR21	P	19.65	12.30	10.58	20.02	14.65	12.61	27.94	17.34	15.24	-	-	-
	MonoRCNN [60]	ICCV21	H	18.36	12.65	10.03	16.61	13.19	10.65	25.48	18.11	14.10	25.29	19.22	15.30
	Autoshape [40]	ICCV21	P	22.47	14.17	11.36	20.09	14.65	12.07	30.66	20.08	15.95	-	-	-
	DCD [35]	ECCV22	P	23.81	15.90	13.21	23.94	17.38	15.32	32.55	21.50	18.25	-	-	-
Hybrid	MonoFlex [80]	CVPR21	EH	19.94	13.89	12.07	23.64	17.51	14.83	28.23	19.75	16.89	-	-	-
	GUPNet [41]	ICCV21	EH	22.26	15.02	13.12	22.76	14.46	13.72	30.29	21.19	18.20	31.07	22.94	19.75
	MonoJSG [40]	CVPR22	EP	24.69	16.14	13.64	26.4	18.3	15.4	32.59	21.26	18.18	-	-	-
	MonoDDE [36]	CVPR22	EHP	24.93	17.14	15.10	26.66	19.75	16.72	33.58	23.46	20.37	35.51	26.48	23.07
	DEVIANT [28]	ECCV22	EH	21.88	14.46	11.89	24.63	16.54	14.52	29.65	20.44	17.43	32.60	23.04	19.99
	NeurOCS-M	CVPR23	EP	**29.80**	**18.60**	**15.62**	**31.31**	**21.07**	**17.79**	**37.50**	**24.39**	**20.77**	**39.26**	**26.91**	**23.69**
	NeurOCS-MLC	CVPR23	EP	**29.89**	**18.94**	**15.90**	**31.24**	**21.01**	**17.70**	**37.27**	**24.49**	**20.89**	**39.16**	**26.78**	**23.63**

NeurOCS-M = Trained w/ Mask, NeurOCS-MLC = Trained w/ Mask+Lidar+LidarComp, **P** = PnP optimization and its variants, **E** = Direct depth estimation, **H** = Depth from height

Table 1. **Comparisons with the state-of-the-arts in KITTI Benchmark,** using AP_{40} with IoU$\geq$0.7 on test and validation set. Note that some methods use depth prediction from DORN [15] whose training data overlaps with the validation set as observed by [7, 70, 71], causing data leakage; these results are marked by blue.

3D boxes and BEV boxes on the ground. The objects are grouped into three difficulty levels - easy, moderate and hard.

Waymo and NuScenes. We follow [28, 37, 68] to train and evaluate on Waymo [64] dataset using its front camera. In addition, we follow [28, 60] to perform cross-dataset evaluation on NuScenes [5]. These results are discussed in the supplementary material in the interest of space.

4.2. Evaluation on KITTI Benchmark

We report the evaluation results in Tab. 1. As discussed in Sec. 2, the existing methods are grouped into three categories according to how they obtain object depth or location, including direct depth regression, geometric methods, and hybrid methods. The geometric cues may arise from PnP-like optimization with sparse or dense keypoints, as well as depth from height or edges. We report the results from NeurOCS with scale fusion with the direct depth regression from the base 3D detector [55]. Also, we report results both when our shape model is trained with instance masks as the major shape cue without using Lidar (NeurOCS-M), and when Lidar and its completion are also applied for shape supervision (NeurOCS-MLC). We do not compare here to the orthogonal works [22, 54] that mainly rely on extra unlabeled data for improvements, leaving the discussion to supp. material. As per common practice, the comparisons are primarily on the test set where the evaluation is done by KITTI servers using withheld ground truth, although we also provide results on validation set for a reference. As can be seen, NeurOCS achieves state-of-the-art accuracy superior to existing methods with a large margin. In addition, NeurOCS-M yields strong performance close to or superior to NeurOCS-MLC. This demonstrates the power of our shape model that effectively translates mask annotations into NOCS supervision. We provide qualitative examples of our results in Fig. 5 to demonstrate its effectiveness; more results running on entire sequences are shown in supplementary.

4.3. Analysis and Ablation Study

Next, we conduct various analyses and ablation studies on validation set to understand the behavior of NeurOCS.

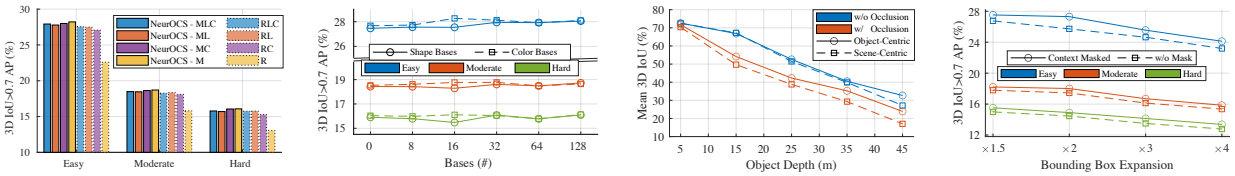
4.3.1 The Role of NeRF

We study the impact of NeRF under different supervision.

w/ NeRF. In addition to NeurOCS-M and NeurOCS-MLC discussed in Sec. 4.2, we also evaluate NeurOCS-ML and NeurOCS-MC, that add Lidar or Lidar completion as supervision to NeurOCS-M.

w/o NeRF. Then, we remove NeRF and train the NOCS prediction branch directly with raw depth supervision, similarly to [7, 37]. Here, the counterpart to our mask-only supervision is the reprojection error loss proposed by [7], denoted as "R". Again, additional supervision from Lidar, Lidar completion, or both are denoted as "RC", "RL", and "RLC".

We compare performance with the PnP-only solution to remove impact from fusion, as shown in Fig. 4a. First observe that "NeurOCS-M" outperforms "R" by a significant margin. This indicates the effectiveness of our method in leveraging instance masks as a weak supervision for NOCS learning even without Lidar. Conversely, reprojection er-



(a) Impact of NeRF (b) Deformation Bases (c) Object-Centric v.s. Scene-Centric (d) Impact of Object-centric Focus

Figure 4. **Performance analysis** of (a) Impact of NeRF, (b) deformation bases, (b)(c) object-centric focus. The abbreviation look-up in Fig. 4a is **L**=Lidar, **C**=Lidar Completion, **M**=Mask, **R**=Reprojection Loss. All results are from the PnP solution without fusion.

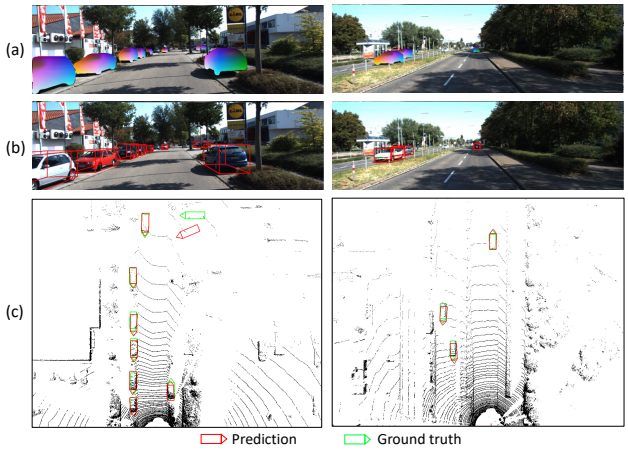


Figure 5. **Qualitative examples** of our method with (a) predicted NOCS and (b) 3D boxes; (c) BEV boxes are also plotted on Lidar point cloud with the ground truth.

		PnP Only			+ Fusion		
		Easy	Moderate	Hard	Easy	Moderate	Hard
Supervision	NeurOCS-MLC	27.92	18.49	15.78	31.24	21.01	17.70
	NeurOCS-ML	27.78	18.45	15.70	31.07	20.94	17.72
	NeurOCS-MC	28.00	18.63	16.06	31.27	21.08	17.79
	NeurOCS-M	28.22	18.71	16.08	31.31	21.07	17.79
NeurOCS-MLC	w/o Deform Reg.	26.33	17.68	14.71	29.69	20.27	17.06
	w/o DetScore	23.48	16.84	14.94	27.20	19.54	17.06
	w/o DetScore & Jac.	20.02	14.61	13.39	22.77	16.84	15.18
	w/o T.T.A.	27.54	18.28	15.42	31.02	20.92	17.57
	Scene-centric	24.55	16.58	13.78	28.30	19.61	16.55
	Directly regress deform.	27.18	18.02	14.88	30.29	20.54	17.18
	Off-the-shelf Mask	27.84	18.56	15.60	30.86	20.87	17.63
NeurOCS-M	w/o Dense Prior	28.17	18.76	16.08	31.18	21.00	17.79
	Off-the-shelf Mask	27.74	18.74	15.70	30.89	20.98	17.49

Table 2. **Ablation study** of various design choices using AP_{3D}.

rors impose weaker shape constraints as the depth ambiguity along the viewing ray persists, which does affect the downstream 3D localization. The usage of Lidar or its completion largely improves the performance when without using NeRF. But the NeRF-based shape model results in superior accuracy by serving as a bridge between the raw Lidar data and the NOCS network that improves the supervision quality.

Next, we demonstrate qualitative examples of different supervisions in the form of point clouds, shown in Fig. 6. The corresponding NOCS regression outputs under different supervisions are also visualized. One observes that our use of NeRF induces high-quality dense NOCS supervsion and hence prediction, in comparison to sparse Lidar and its completion. Remarkably, our weakly-supervised NOCS learning with instance masks yields far better shape than the self-supervised reprojection error loss.

4.3.2 Key Design Choices

Next, we study the impact of various design choices in Tab. 2 and Fig. 4(b)-(d), using the validation set.

Supervision sources. We first report in Tab. 2 the results under different sources of supervision, including NeurOCS-MLC, NeurOCS-ML, NeurOCS-MC, and NeurOCS-M, where "M", "L" and "C" respectively denote supervision from instance mask, Lidar, and Lidar completion. We observe competitive performance across all the four settings, indicating the robustness of our method, and importantly the capability of our shape model in effectively translating instance mask into high-quality NOCS supervision. In addition, our PnP-based solution significantly outperforms existing geometric methods listed in Tab. 1 that primarily rely on geometric reasoning as well. Furthermore, we note that the fusion of PnP with the scale from direct depth regression consistently improves the performance, indicating their complementary nature.

KL regularization. We study the impact of the KL regularization loss on the shape basis coefficients and find it to be important. As shown in Tab. 2 "w/o Deform Reg.", removing this loss from NeurOCS-MLC leads to a large drop in accuracy. This implies server ambiguities in the shape modeling despite the low-rank structure in the latent grid, due to the challenging conditions in real driving scenes.

Deformation bases. We study the performance with varying number of deformation bases for both shape and color space, as shown in Fig. 4b. We observe that our NeRF-rendered NOCS leads to good localization performance even with just the canonical shape model, *i.e.* using zero basis. However, the presence of deformation bases improves the accuracy by accounting for instance-level variations. In practice, we use 64 bases as a trade-off of accuracy and computation.

Directly regressing deformations. Instead of decomposing the latent grid of NeRF as the shared canonical one plus per-instance linear deformation with shared basis, an alternative

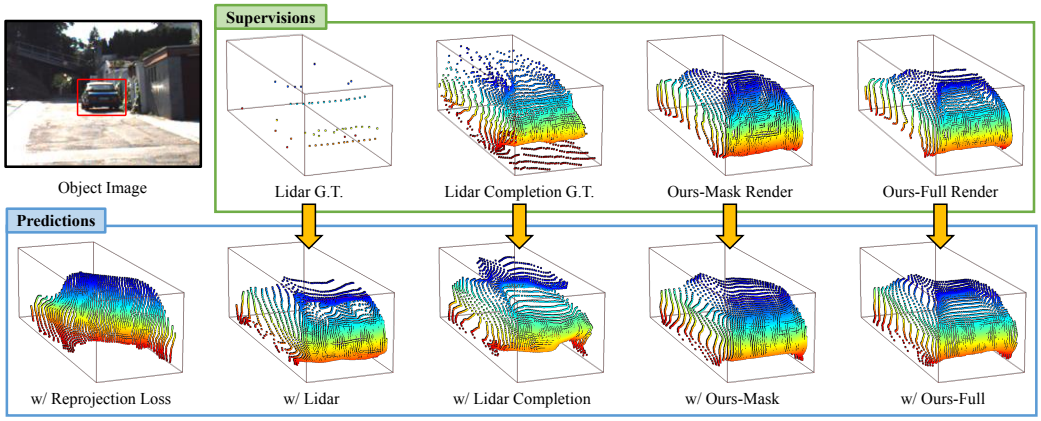


Figure 6. **Qualitative comparison of shapes learnt from different sources of supervision.** We visualize the NOCS by projecting them to point clouds with ground-truth object size. The point cloud is colored with Y-axis coordinate (height).

way is to let the network directly regress the latent grid for each instance independently without any constraints. We observe that this yields degraded performance as shown in Tab. 2, which indicates the benefits of the low-rank inductive bias in categorical shape learning.

Visual scope. We study the behavior of the object-centric and scene-centric training strategies (Sec. 3.4). We follow [7] to use ResNet101 with FPN [38] to handle object scales in scene-centric scheme. Similar results with DLA [77] are in supplementary. We report the AP metric for the scene-centric case in Tab. 2, which shows the default object-centric training is largely superior. The same is observed in supplementary when training with Lidar without NeRF. To understand this benefit, we analyze per-instance 3D IoU w.r.t the ground truth 3D boxes. Specifically, we group all recalled objects according to their ground truth depth, separate objects under occlusions (or truncations) from fully visible ones, and then plot the mean 3D IoU for each group against object depth. The results with direct Lidar training is shown in Fig. 4c. We observe that the object-centric training benefits more to distant and occluded objects – regardless of the occlusions, the performance gap between the two schemes increases as depth increases; and evidently the performance gap is much larger for objects under occlusions. The performance gap remains even with larger network capacity (detailed in supp.)

We further study the benefit of object-centric focus with its slight variant – enlarging the detection boxes before cropping to include more scene context. We evaluate two settings: (a) feeding the object patches to networks as is; (b) masking out the additional context before doing (a). The additional context reduces the degree of object-centric focus, and in this way we isolate its effect. Results with direct Lidar training is shown in Fig. 4d, where context masking-out yields higher accuracy, with a larger gap as the boxes expand more, demonstrating the benefits of a greater object-centric focus.

Detection score and Jacobian map. We first remove the detection score from the base 3D detector (Tab. 2 "w/o DetScore"), and then additionally remove the Jacobian map from our score branch ("w/o DetScore & Jac."). The results show our PnP score itself is meaningful, and the Jacobian map contributes positively to the score prediction.

Test-time augmentation (T.T.A.). We note that accuracy improvement (Tab. 2 "w/o T.T.A.") is brought by T.T.A.

Dense shape prior. In the absence of Lidar, removing the dense prior from NeurOCS-M results in a drop (Tab. 2 "w/o Dense Prior"), indicating the dense prior may mitigate the visual hull ambiguity in single-view shape learning.

Off-the-shelf instance masks. Instead of the ground truth masks, we adopt the mask prediction from an off-the-shelf pretrained Mask R-CNN [18] model. This obviates the need for additional mask annotations in KITTI. We report in Tab. 2 the accuracy for NeurOCS-MLC and NeurOCS-M. While the performance drops as expected, it remains strong and outperforms existing methods shown in Tab. 1. This demonstrates its robustness with respect to instance masks.

5. Limitations and Conclusion

In this work, we propose a 3D localization framework that rests on NOCS-based object pose estimation but leverages NeRF to address its key challenge – the lack of supervision in real driving scenes. We learn category-level neural shape models to provide high-quality NOCS supervision. Driven by crucial design choices for effective NOCS learning and ambiguity handling, our method yields new state-of-the-art performance. One limitation of our method lies in its reliance on a base 3D detector and its object size prediction. Also, vehicles with irregular shape may cause challenges to our NOCS prediction. These remain interesting problems to explore in the future. Further, we envision our work to inspire more efforts towards further unleashing the potential of differentiable rendering for 3D object localization.

References

[1] Wentao Bao, Bin Xu, and Zhenzhong Chen. Monofenet: Monocular 3d object detection with feature enhancement networks. *TIP*, 2019. 2

[2] Deniz Beker, Hiroharu Kato, Mihai Adrian Morariu, Takahiro Ando, Toru Matsuoka, Wadim Kehl, and Adrien Gaidon. Monocular differentiable rendering for self-supervised 3d object detection. In *ECCV*, 2020. 1, 2

[3] Garrick Brazil and Xiaoming Liu. M3d-rpn: Monocular 3d region proposal network for object detection. In *ICCV*, 2019. 2

[4] Garrick Brazil, Gerard Pons-Moll, Xiaoming Liu, and Bernt Schiele. Kinematic 3d object detection in monocular video. In *ECCV*, 2020. 1

[5] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *CVPR*, 2020. 6

[6] 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 *CVPR*, 2017. 2

[7] Hansheng Chen, Yuyao Huang, Wei Tian, Zhong Gao, and Lu Xiong. Monorun: Monocular 3d object detection by reconstruction and uncertainty propagation. In *CVPR*, 2021. 2, 5, 6, 8

[8] Hansheng Chen, Pichao Wang, Fan Wang, Wei Tian, Lu Xiong, and Hao Li. Epro-pnp: Generalized end-to-end probabilistic perspective-n-points for monocular object pose estimation. In *CVPR*, 2022. 2, 4

[9] Xiaozhi Chen, Kaustav Kundu, Ziyu Zhang, Huimin Ma, Sanja Fidler, and Raquel Urtasun. Monocular 3d object detection for autonomous driving. In *CVPR*, 2016. 2

[10] Xiaozhi Chen, Kaustav Kundu, Yukun Zhu, Andrew G Berneshawi, Huimin Ma, Sanja Fidler, and Raquel Urtasun. 3d object proposals for accurate object class detection. *NeurIPS*, 2015. 5

[11] Yi-Nan Chen, Hang Dai, and Yong Ding. Pseudo-stereo for monocular 3d object detection in autonomous driving. In *CVPR*, 2022. 6

[12] Kong-man German Cheung, Simon Baker, and Takeo Kanade. Shape-from-silhouette across time part i: Theory and algorithms. *ICCV*, 2005. 4

[13] 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 *CVPR Workshops*, 2020. 2

[14] Wolfgang Frstner and Bernhard P. Wrobel. *Photogrammetric Computer Vision: Statistics, Geometry, Orientation and Reconstruction*. Springer Publishing Company, Incorporated, 2016. 5

[15] Huan Fu, Mingming Gong, Chaohui Wang, Kayhan Batmanghelich, and Dacheng Tao. Deep Ordinal Regression Network for Monocular Depth Estimation. In *CVPR*. 6

[16] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *CVPR*, 2012. 1, 2, 5

[17] Jiaqi Gu, Bojian Wu, Lubin Fan, Jianqiang Huang, Shen Cao, Zhiyu Xiang, and Xian-Sheng Hua. Homography loss for monocular 3d object detection. In *CVPR*, 2022. 2, 6

[18] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, 2017. 8

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 3

[20] Jonas Heylen, Mark De Wolf, Bruno Dawagne, Marc Proesmans, Luc Van Gool, Wim Abbeloos, Hazem Abdelkawy, and Daniel Olmeda Reino. Monocinis: Camera independent monocular 3d object detection using instance segmentation. In *ICCV*, 2021. 5

[21] Tomáš Hodaň, Martin Sundermeyer, Bertram Drost, Yann Labbé, Eric Brachmann, Frank Michel, Carsten Rother, and Jiří Matas. BOP challenge 2020 on 6D object localization. *ECCV Workshops*, 2020. 1

[22] Yu Hong, Hang Dai, and Yong Ding. Cross-modality knowledge distillation network for monocular 3d object detection. In *ECCV*, 2022. 2, 6

[23] Mu Hu, Shuling Wang, Bin Li, Shiyu Ning, Li Fan, and Xiaojin Gong. Penet: Towards precise and efficient image guided depth completion. In *ICRA*, 2021. 1, 4

[24] Kuan-Chih Huang, Tsung-Han Wu, Hung-Ting Su, and Winston H Hsu. Monodtr: Monocular 3d object detection with depth-aware transformer. In *CVPR*, 2022. 2, 6

[25] Longlong Jing, Ruichi Yu, Henrik Kretzschmar, Kang Li, Charles R Qi, Hang Zhao, Alper Ayvaci, Xu Chen, Dillon Cower, Yingwei Li, et al. Depth estimation matters most: Improving per-object depth estimation for monocular 3d detection and tracking. *arXiv preprint arXiv:2206.03666*, 2022. 2

[26] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 2, 4

[27] Jason Ku, Alex D. Pon, and Steven L. Waslander. Monocular 3d object detection leveraging accurate proposals and shape reconstruction. In *CVPR*, 2019. 2

[28] Abhinav Kumar, Garrick Brazil, Enrique Corona, Armin Parchami, and Xiaoming Liu. DEVIANT: Depth EquiVarIAnt NeTwork for Monocular 3D Object Detection. In *ECCV*, 2022. 2, 6

[29] Abhinav Kumar, Garrick Brazil, and Xiaoming Liu. GrooMeD-NMS: Grouped mathematically differentiable nms for monocular 3D object detection. In *CVPR*, 2021. 6

[30] Abhijit Kundu, Kyle Genova, Xiaoqi Yin, Alireza Fathi, Caroline Pantofaru, Leonidas J. Guibas, Andrea Tagliasacchi, Frank Dellaert, and Thomas Funkhouser. Panoptic neural fields: A semantic object-aware neural scene representation. In *CVPR*, 2022. 3

[31] Abhijit Kundu, Yin Li, and James M. Rehg. 3d-rcnn: Instance-level 3d object reconstruction via render-and-compare. In *CVPR*, 2018. 1, 2

[32] Alex H Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *CVPR*, 2019. 1

[33] Vincent Lepetit, Francesc Moreno-Noguer, and Pascal Fua. Epnp: An accurate o (n) solution to the pnp problem. *IJCV*, 2009. 1

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

[35] Yingyan Li, Yuntao Chen, Jiawei He, and Zhaoxiang Zhang. Densely constrained depth estimator for monocular 3d object detection. In *ECCV*, 2022. 1, 2, 6

[36] Zhuoling Li, Zhan Qu, Yang Zhou, Jianzhuang Liu, Haoqian Wang, and Lihui Jiang. Diversity matters: Fully exploiting depth clues for reliable monocular 3d object detection. In *CVPR*, 2022. 2, 5, 6

[37] Qing Lian, Peiliang Li, and Xiaozhi Chen. Monojsg: Joint semantic and geometric cost volume for monocular 3d object detection. In *CVPR*, 2022. 2, 5, 6

[38] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017. 8

[39] Yuxuan Liu, Yuan Yixuan, and Ming Liu. Ground-aware monocular 3d object detection for autonomous driving. *IEEE Robotics and Automation Letters*, 6(2):919–926, 2021. 2

[40] Zongdai Liu, Dingfu Zhou, Feixiang Lu, Jin Fang, and Liangjun Zhang. Autoshape: Real-time shape-aware monocular 3d object detection. In *ICCV*, 2021. 2, 6

[41] Yan Lu, Xinzhu Ma, Lei Yang, Tianzhu Zhang, Yating Liu, Qi Chu, Junjie Yan, and Wanli Ouyang. Geometry uncertainty projection network for monocular 3d object detection. In *ICCV*, 2021. 2, 6

[42] Xinzhu Ma, Shinan Liu, Zhiyi Xia, Hongwen Zhang, Xingyu Zeng, and Wanli Ouyang. Rethinking pseudo-lidar representation. In *ECCV*, 2020. 2

[43] 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 *CVPR*, 2021. 1, 2, 3, 6

[44] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 2, 3, 4

[45] Zhixiang Min and Enrique Dunn. Voldor+ slam: For the times when feature-based or direct methods are not good enough. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 13813–13819. IEEE, 2021. 2

[46] Zhixiang Min, Naji Khosravan, Zachary Bessinger, Manjunath Narayana, Sing Bing Kang, Enrique Dunn, and Ivaylo Boyadzhiev. Laser: Latent space rendering for 2d visual localization. In *CVPR*, 2022. 2

[47] Zhixiang Min, Yiding Yang, and Enrique Dunn. Voldor: Visual odometry from log-logistic dense optical flow residuals. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4898–4909, 2020. 2

[48] Arsalan Mousavian, Dragomir Anguelov, John Flynn, and Jana Kosecka. 3d bounding box estimation using deep learning and geometry. In *CVPR*, 2017. 2

[49] Norman Müller, Andrea Simonelli, Lorenzo Porzi, Samuel Rota Bulò, Matthias Nießner, and Peter Kontschieder. Autorf: Learning 3d object radiance fields from single view observations. In *CVPR*, 2022. 3, 4

[50] Thomas Müller. tiny-cuda-nn, 4 2021. 2, 3

[51] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Trans. Graph.*, 41(4):102:1–102:15, July 2022. 2, 3

[52] Julian Ost, Fahim Mannan, Nils Thuerey, Julian Knodt, and Felix Heide. Neural scene graphs for dynamic scenes. In *CVPR*, 2021. 3

[53] Dennis Park, Rares Ambrus, Vitor Guizilini, Jie Li, and Adrien Gaidon. Is pseudo-lidar needed for monocular 3d object detection? In *ICCV*, 2021. 2

[54] Liang Peng, Fei Liu, Zhengxu Yu, Senbo Yan, Dan Deng, Zheng Yang, Haifeng Liu, and Deng Cai. Lidar point cloud guided monocular 3d object detection. In *ECCV*, 2022. 2, 6

[55] Liang Peng, Xiaopei Wu, Zheng Yang, Haifeng Liu, and Deng Cai. Did-m3d: Decoupling instance depth for monocular 3d object detection. *arXiv preprint arXiv:2207.08531*, 2022. 2, 3, 5, 6

[56] 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 *CVPR*, 2020. 2

[57] Zequn Qin and Xi Li. Monoground: Detecting monocular 3d objects from the ground. In *CVPR*, 2022. 2, 6

[58] Cody Reading, Ali Harakeh, Julia Chae, and Steven L Waslander. Categorical depth distribution network for monocular 3d object detection. In *CVPR*, 2021. 2, 6

[59] Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. Pointrcnn: 3d object proposal generation and detection from point cloud. In *CVPR*, 2019. 1

[60] Xuepeng Shi, Qi Ye, Xiaozhi Chen, Chuangrong Chen, Zhixiang Chen, and Tae-Kyun Kim. Geometry-based distance decomposition for monocular 3d object detection. In *ICCV*, 2021. 2, 6

[61] Andrea Simonelli, Samuel Rota Bulo, Lorenzo Porzi, Peter Kontschieder, and Elisa Ricci. Are we missing confidence in pseudo-lidar methods for monocular 3d object detection? In *ICCV*, 2021. 2

[62] Andrea Simonelli, Samuel Rota Bulo, Lorenzo Porzi, Manuel López-Antequera, and Peter Kontschieder. Disentangling monocular 3d object detection. In *ICCV*, 2019. 2, 5

[63] Krishna Kumar Singh, Dhruv Mahajan, Kristen Grauman, Yong Jae Lee, Matt Feiszli, and Deepti Ghadiyaram. Don't judge an object by its context: Learning to overcome contextual bias. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11070–11078, 2020. 5

[64] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, Vijay Vasudevan, Wei Han, Jiquan Ngiam, Hang Zhao, Aleksei Timofeev, Scott Ettinger, Maxim Krivokon, Amy Gao, Aditya Joshi, Yu Zhang, Jonathon Shlens, Zhifeng Chen, and Dragomir Anguelov.

Scalability in perception for autonomous driving: Waymo open dataset. In *CVPR*, 2020. 6

[65] Ayush Tewari, Justus Thies, Ben Mildenhall, Pratul Srinivasan, Edgar Tretschk, W Yifan, Christoph Lassner, Vincent Sitzmann, Ricardo Martin-Brualla, Stephen Lombardi, et al. Advances in neural rendering. In *Computer Graphics Forum*, volume 41, pages 703–735. Wiley Online Library, 2022. 1

[66] He Wang, Srinath Sridhar, Jingwei Huang, Julien Valentin, Shuran Song, and Leonidas J Guibas. Normalized object coordinate space for category-level 6d object pose and size estimation. In *CVPR*, 2019. 1, 2

[67] 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 *CVPR*, 2021. 2, 6

[68] Li Wang, Li Zhang, Yi Zhu, Zhi Zhang, Tong He, Mu Li, and Xiangyang Xue. Progressive coordinate transforms for monocular 3d object detection. *NeurIPS*, 2021. 2, 6

[69] Tai Wang, Jiangmiao Pang, and Dahua Lin. Monocular 3d object detection with depth from motion. In *ECCV*, 2022. 2

[70] Xinlong Wang, Wei Yin, Tao Kong, Yuning Jiang, Lei Li, and Chunhua Shen. Task-aware monocular depth estimation for 3d object detection. In *AAAI*, 2020. 6

[71] 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 *CVPR*, 2019. 2, 6

[72] Xiaopei Wu, Liang Peng, Honghui Yang, Liang Xie, Chenxi Huang, Chengqi Deng, Haifeng Liu, and Deng Cai. Sparse fuse dense: Towards high quality 3d detection with depth completion. In *CVPR*, 2022. 1

[73] Lei Yang, Xinyu Zhang, Li Wang, Minghan Zhu, Chuang Zhang, and Jun Li. Mix-teaching: A simple, unified and effective semi-supervised learning framework for monocular 3d object detection. *arXiv preprint arXiv:2207.04448*, 2022. 2

[74] Qian Ye, Ling Jiang, and Yuyang Du. Consistency of implicit and explicit features matters for monocular 3d object detection. *arXiv preprint arXiv:2207.07933*, 2022. 2

[75] 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. 2

[76] Alex Yu, Ruilong Li, Matthew Tancik, Hao Li, Ren Ng, and Angjoo Kanazawa. PlenOctrees for real-time rendering of neural radiance fields. In *ICCV*, 2021. 4

[77] Fisher Yu, Dequan Wang, Evan Shelhamer, and Trevor Darrell. Deep layer aggregation. In *CVPR*, 2018. 8

[78] Sergey Zakharov, Rares Andrei Ambrus, Vitor Campagnolo Guizilini, Dennis Park, Wadim Kehl, Fredo Durand, Joshua B Tenenbaum, Vincent Sitzmann, Jiajun Wu, and Adrien Gaidon. Single-shot scene reconstruction. In *5th Annual Conference on Robot Learning*, 2021. 1, 2

[79] Sergey Zakharov, Wadim Kehl, Arjun Bhargava, and Adrien Gaidon. Autolabeling 3d objects with differentiable rendering of sdf shape priors. In *CVPR*, 2020. 1, 2, 5

[80] Yunpeng Zhang, Jiwen Lu, and Jie Zhou. Objects are different: Flexible monocular 3d object detection. In *CVPR*, 2021. 2, 6

[81] Yunpeng Zhang, Wenzhao Zheng, Zheng Zhu, Guan Huang, Dalong Du, Jie Zhou, and Jiwen Lu. Dimension embeddings for monocular 3d object detection. In *CVPR*, 2022. 2

[82] Yunsong Zhou, Yuan He, Hongzi Zhu, Cheng Wang, Hongyang Li, and Qinhong Jiang. Monocular 3d object detection: An extrinsic parameter free approach. In *CVPR*, 2021. 2, 6

[83] Yunsong Zhou, Quan Liu, Hongzi Zhu, Yunzhe Li, Shan Chang, and Minyi Guo. Mogde: Boosting mobile monocular 3d object detection with ground depth estimation. In *NeurIPS*, 2022. 2