BAAM: Monocular 3D pose and shape reconstruction with bi-contextual attention module and attention-guided modeling

Hyo-Jun Lee[1†] Hanul Kim[3†] Su-Min Choi[2] Seong-Gyun Jeong[2] Yeong Jun Koh[1*]
[1]Chungnam National University [2]42dot Inc.
[3]Seoul National University of Science and Technology
gywns6287@gmail.com, hukim@seoultech.ac.kr, sumin.choi@42dot.ai,
seonggyun.jeong@42dot.ai, yjkoh@cnu.ac.kr

Abstract

3D traffic scene comprises various 3D information about car objects, including their pose and shape. However, most recent studies pay relatively less attention to reconstructing detailed shapes. Furthermore, most of them treat each 3D object as an independent one, resulting in losses of relative context inter-objects and scene context reflecting road circumstances. A novel monocular 3D pose and shape reconstruction algorithm, based on bi-contextual attention and attention-guided modeling (BAAM), is proposed in this work. First, given 2D primitives, we reconstruct 3D object shape based on attention-guided modeling that considers the relevance between detected objects and vehicle shape priors. Next, we estimate 3D object pose through bi-contextual attention, which leverages relation-context inter objects and scene-context between an object and road environment. Finally, we propose a 3D non-maximum suppression algorithm to eliminate spurious objects based on their Bird-Eye-View distance. Extensive experiments demonstrate that the proposed BAAM yields state-of-the-art performance on ApolloCar3D. Also, they show that the proposed BAAM can be plugged into any mature monocular 3D object detector on KITTI and significantly boost their performance. Code is available at https://github.com/gywns6287/BAAM.

1. Introduction

3D traffic scene understanding provides enriched descriptions of the dynamic objects, *e.g.*, 3D shape, pose, and location, compared to representing objects as bounding boxes. 3D visual perception is crucial for the autonomous driving system to develop downstream tasks such as motion prediction and planning, and aids to faithfully reconstruct the traffic scene from recorded data. To acquire precise 3D information, some prior arts have relied on specific devices such as LiDAR [3,10,42] and stereo vision [26,44]. However, as the system becomes complex and costly, it quickly reaches the limit to scalability. To contrary, areas of study about 3D perception using monocular vision have been receiving attention due to its simplicity and cost efficiency [4,7,19,29,33,50,51].

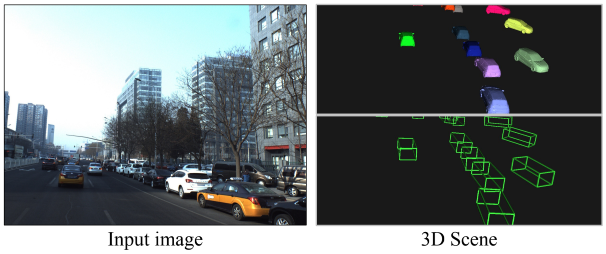

Figure 1. Reconstructed 3D scene with rough bounding box (right up) and with detailed shape (right down). For better 3D reconstruction, detailed 3D shapes are needed rather than the simple 3D bounding boxes.

Monocular 3D perception is an ill-posed problem in that projective geometry inherently loses depth information. In particular, traffic scene contains partially observable objects, and shows fine-grained classes which are visually confusing. Pseudo-LiDAR [49] presents a feasible solution of the image based 3D object detection. To reconstruct 3D poses of the objects, many studies [25,27,30,33,40,41,50,51] focus on using geometry constraints between 2D and 3D. Yet, it is less studied in the line of research that leverage relative context among the objects and global scene context depending on road environment.

Figure 1 compares the reconstructed 3D scene with 3D bounding boxes and detailed 3D shapes. With a detailed 3D shape, we render the traffic scene in realistic and provide intuitive representations of the objects. Despite scale ambiguity of the monocular 3D perception, 3D mesh provides a

[*] Corresponding author
[†] These authors contributed equally to this work.

strong clue to align instances' scales and orientations. Concurrently, there have been many attempts [8, 20, 22, 31, 45, 46] to reconstruct the 3D shape of human objects. These methods mainly focus on learning PCA-basis to represent human shapes. Inspired by human shape reconstruction, recent methods [21, 24] also design PCA-basis for vehicle shape reconstruction. However, as pointed out in [1, 34], PCA-basis often loses object details and thus leads to unsatisfactory reconstruction.

In this work, we propose a novel 3D pose and shape estimation algorithm, utilizing bi-contextual attention and attention-guided modeling (BAAM). Given a monocular RGB image, the proposed BAAM first extracts various 2D primitive features such as appearance, position, and size. And it constructs object features to embed internal object structures by aggregating primitive features. For detailed object shapes, we introduce shape priors consisting of the mean shape and various template offsets to represent details of vehicle shapes. Then, BAAM reconstructs objects' 3D shapes as mesh structures with attention-guided modeling, which combines shape prior and individual object features based on their relevance. For accurate pose estimation, we present the notion of bi-contextual attention consisting of relation-context and scene-context, which describe the relationship inter objects and between object and road environment, respectively. Based on this rich information, BAAM integrates object features to predict objects' 3D poses through a carefully designed bi-contextual attention module. Finally, we proposed a novel 3D non-maximum suppression (NMS) algorithm that effectually removes spurious objects based on Bird-Eye-view (BEV) geometry. Extensive experiments on Apollocar3D [43] and KITTI [12] datasets demonstrate the effectiveness of the proposed BAAM algorithm. Also, experiments show that the proposed method significantly outperforms state-of-the arts [21, 43] in both pose and shape estimation. The main contributions of our work are four folds:

- We propose the attention-guided modeling that reconstructs objects' shapes based on the relevance between objects and vehicle shape priors.
- We proposed the bi-contextual attention module that estimates objects' pose by exploiting relation-context inter objects and scene-context between an object and road environment.
- We also develop the novel 3D non-maximum suppression algorithm to remove spurious objects based on their Bird-Eye-view distance.
- The proposed BAAM algorithm achieves the state-of the art performance on ApolloCar3D [43]. Also, experiments on KITTI [12] show that the proposed algorithm can significantly improve the performance of existing monocular 3D detectors.

2. Related Work

Monocular 3D object detection. Monocular 3D object detection aims to estimate 3D bounding boxes of objects in a given image. Existing monocular 3D object detection methods are roughly categorized into depth-assisted and image-only methods. The depth-assisted approach uses a pixel-wise depth map to aid 3D object detection by training a monocular depth estimator. Pseudo-LiDAR [36, 49] methods transform estimated depth maps into 3D point clouds and feed them into the existing LiDAR-based 3D detectors. PatchNet [35] takes advantage of CNNs by representing transformed 3D information in images. DDMP-3D designs the message passing block to transfer depth information to 3D detectors. DD3D [39] pre-train depth estimator and fine-tune it for 3D object detection. In [40], DID-M3D decouples object depth into visual and attribute depth. Although these depth-assisted methods [35,36,39,40,49] have the advantage of improving 3D detection quality, they have limitations in that they require additional information.

Due to the lack of depth information, many image-only methods focus on exploiting geometry priors. Deep3DBox [38] introduces MultiBin loss for rotation estimation and solves the translation by geometrical relationship between 2D-3D boxes. GUPNet [33] combines geometry priors and uncertainty modeling to infer depth distribution. MonoFlex [50] decouples truncated objects and formulates the depth estimation as an uncertainty-guided depth ensemble. MonoGround [41] introduced the notion of ground plane to convert the ill-posed 2D to 3D mapping into a well-posed problem with a unique solution. In [51], Zhang *et al*. developed the dimension-aware embedding for more accurate geometric constraints. However, these methods are limited in that they do not fully consider rich context in monocular images, which gives additional cues for depth estimation. In contrast, MonoPair [7] adopts the pair-wise relationship between neighboring objects to post-optimize object translation. In [14], Gu *et al*. presented the homography loss to constrain mutual locations of objects in the 3D scene. Both methods concentrate on the geometrical association between predicted objects. On the contrary, the proposed BAAM algorithm considers object relation in feature space, which implies various cues for 3D pose.

Monocular 3D pose and shape reconstruction. Joint 3D pose and shape regression has been actively studied for human objects [8, 20, 22, 31, 45, 46]. Inspired by human shape reconstruction, there have been many attempts to restore vehicle shape for 3D traffic scene understanding. DeepMANTA [5] adopts a coarse-to-fine retrieval strategy to reconstruct pose and skeleton shape. 3D-RCNN [24] and Roi-10D [37] render 3D shapes as PCA parameters, which are decoded to coarse voxel shape representation. In [43], the direct-based approach extends 3D-RCNN to utilize attention mask and offset flow. GSNet [21] presents a divide-

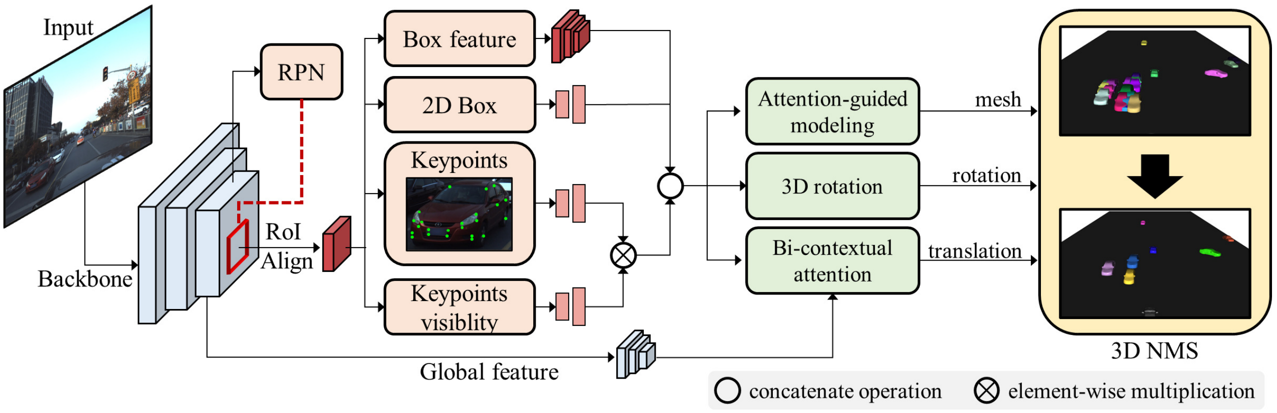

Figure 2. Overview of the proposed BAAM. The input image is sent to the network to extract box features, 2D box, and keypoints. Given, the informative 2D representation, BAAM estimates the shapes using attention-guided modeling. 3D rotation is directly regressed by the fully connected layers and 3D translation is faithfully predicted through the bi-contextual attention module.

and-conquer strategy, which generates 3D shapes by blending multiple meshes from different PCA-basis. However, the PCA-basis often lose the details of object shape [1, 34]. To overcome these limitations, we construct shape priors to hold these details. We then propose an attention-guided modeling, which can reconstruct object shapes adopting relevance between object and shape prior.

3. Methods

3.1. Problem Statement

Suppose that there exist n objects in an RGB image. We define each object pose as 3D translation $\mathbf{p}_\mathrm{t} = \{x, y, z\}$ and 3D rotation $\mathbf{p}_\mathrm{r} = \{\alpha, \beta, \gamma\}$. Also, we represent an object shape as 3D mesh $\mathbf{m} \in \mathbb{R}^{3v}$ in [21], where v is the number of vertices, and each vertex has its 3D coordinates. Therefore, a monocular 3D pose and shape reconstruction aims to estimate all objects' translation $\mathbf{P}_\mathrm{t} \in \mathbb{R}^{n\times 3}$, rotation $\mathbf{P}_\mathrm{r} \in \mathbb{R}^{n\times 3}$, and their shape $\mathbf{M} \in \mathbb{R}^{n\times 3v}$.

3.2. Model Overview

Figure 2 illustrates the overall architecture of BAAM. The proposed BAAM extends Mask R-CNN [16] to extract 2D primitive features: bounding box (region of interest, RoI), bounding box feature, keypoints, and keypoint visibility. Given primitive 2D features, it first conveys visibility information to keypoints. Then, it concatenates three types of features to construct object features for 3D pose and shape estimation. Also, it extracts the global feature to encode scene context indicating the road environment. After that, the proposed BAAM predicts objects' poses and shapes based on global and object features. First, it uses attention-guided modeling (AGM) which leverages the relevance between objects and the shape prior about vehicles to estimate object shapes. Second, it faithfully predicts object translation through bi-contextual attention (BCA) including relation-aware attention between each object and scene-aware attention between objects and the road scene. Third, it takes object features to regress their rotation.

3.3. 2D Feature Construction

Object feature. A bounding box denotes object center coordinates (b_x, b_y) and its width b_w and height b_h in image space. Since image formation is closely related to camera parameters [15], it is essential to aggregate camera information into bounding boxes for 3D pose and shape estimation. Hence, we transform b_x, b_y, b_w, b_h from image space to $\tilde{b}_x$, $\tilde{b}_y$, $\tilde{b}_w$, $\tilde{b}_h$ in camera space:

$$\tilde{b}_x = \frac{b_x - c_x}{f_x}, \quad \tilde{b}_y = \frac{b_y - c_y}{f_y}, \quad \tilde{b}_w = \frac{b_w}{f_x}, \quad \tilde{b}_h = \frac{b_h}{f_y} \qquad (1)$$

where f_x, f_y are camera focal lengths and c_x, c_y are camera principal points. We then feed the transformed boxes into two fully-connected layers. A bounding box feature, obtained using RoIAlign [16], contains useful information about object appearance. We further process this feature using three convolution operations to reduce its spatial resolution for the object feature construction.

We estimate keypoint coordinates and visibility for each object defined in [43]. Here, the visibility of each point is a Bernoulli variable that indicates whether that point is visible in the input image. Since keypoints are in image space, we transform them into camera space in the same way as Equation (1), and process them using two different fully-connected layers. Then, we multiply their elements to aggregate visibility information to keypoints. Note that keypoints give not only location and size cues but also shape cues, which are essential in shape estimation, because they involve local structural information of vehicles. Finally, we construct object features $\mathbf{X}_\mathrm{o} \in \mathbb{R}^{n\times c}$ by concatenating bounding boxes, bounding box features, and keypoints,

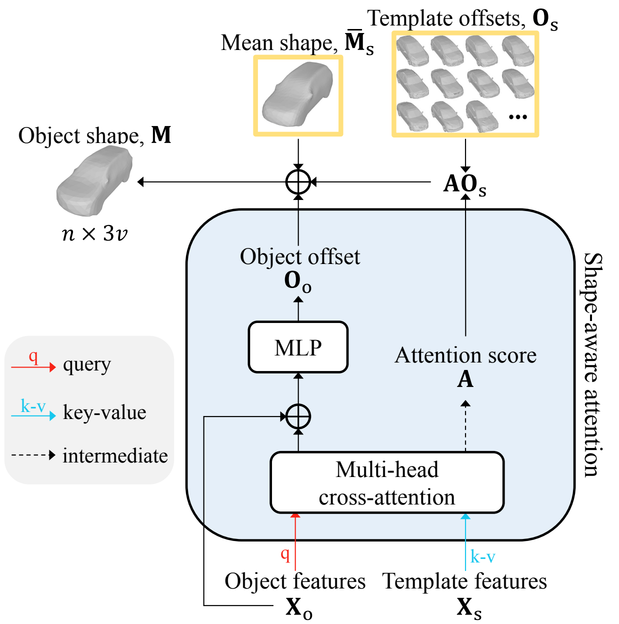

Figure 3. Attention-guided modeling for 3D shape estimation. The shape-aware attention can explore the relevance between the object features and learnable template features to estimate object offset and to generate attention score.

where c is concatenated feature dimension. As a combination of primitive features, object features can provide rich 2D information including shape, position, and size about individual objects for faithful 3D estimation.

Global feature. Different from object features, which describe individual objects, global features represent various scene contexts. Specifically, we feed backbone feature maps into three convolutional layers. We then sequentially apply global average pooling and reshape operator to construct global features $\mathbf{X}_g \in \mathbb{R}^{g\times c}$. Here, the number of global contexts g is fixed to 8.

3.4. Attention-guided modeling

Shape prior. Since a vehicle is a rigid object, we take advantage of prior knowledge about vehicle shapes for 3D shape estimation. To build shape prior, we adopt $p = 79$ mesh templates in [21], in which each template is composed of $v = 1352$ vertices to describe a representative vehicle shape in ApolloCar3D dataset [43]. As shown in Figure 3, we divide them into the mean shape $\bar{\mathbf{m}}_s \in \mathbb{R}^{3v}$ and template offsets $\mathbf{O}_s \in \mathbb{R}^{p\times 3v}$ indicating the difference between the mean shape and templates.

Shape-aware attention. We decompose an object shape into three components: mean shape, template offsets, and object offset. Let $\bar{\mathbf{M}}_s \in \mathbb{R}^{n\times 3v}$ be the mean shape matrix, whose rows are $\bar{\mathbf{m}}_s$, and $\mathbf{O}_o \in \mathbb{R}^{n\times 3v}$ be the object offset matrix, whose rows contain offset for each object. Then We define all objects' shapes as

$$\mathbf{M} = \bar{\mathbf{M}}_s + \mathbf{AO}_s + \mathbf{O}_o \tag{2}$$

where $\mathbf{A} \in \mathbb{R}^{n\times p}$ denotes all objects' attention scores, of which rows indicate templates' contribution to representing each object shape. Therefore, the shape-aware attention aims to estimate all objects' attention scores $\mathbf{A}$ and object offsets $\mathbf{O}_o \in \mathbb{R}^{n\times 3v}$ by exploiting the relationship between objects and shape priors. This decomposition simplifies the original problem, which should directly determine vertex coordinates. Note that since cars are rigid bodies, the range of possible object offsets is limited.

As shown in Figure 3, we first represent template offsets to template features $\mathbf{X}_s \in \mathbb{R}^{p\times c}$ using the standard learnable embedding scheme [9, 48], to measure relevance between an object and templates. We then predict object offsets using multi-head cross-attention (MCA). MCA's mechanism is similar to standard multi-head self-attention (MSA) [48]. The only difference is that MCA takes queries and key-value pairs from different sources. Specifically, we project object features to queries and template features into keys and values. Thus, object offsets are given by

$$\tilde{\mathbf{O}}_o = \mathbf{X}_o + \mathsf{MCA}(\mathsf{LN}(\mathbf{X}_o), \mathbf{X}_s) \tag{3}$$

$$\mathbf{O}_o = \mathsf{MLP}(\tilde{\mathbf{O}}_o) \tag{4}$$

where $\mathsf{LN}(\cdot)$ is a layer normalization [2], $\mathsf{MLP}(\cdot)$ contains two fully-connected layers with GELU non-linearity [18]. Note that attention scores $\mathbf{A}$, which present the similarities between objects and shape priors, are available in the MCA block for object offsets.

3.5. Bi-contextual attention

Given the object features $\mathbf{X}_o$, we directly regress it to the 3D rotation $\mathbf{P}_r$ with 3 fully connected layers. This is because the object feature encodes the internal object structure giving meaningful cues for rotation estimation. On the other hand, it is not straightforward to restore 3D translation from the object feature. Note that 2D images have already lost depth due to the image formation process. Thus, we attempt to use an external object structure to compensate for it. Figure 4 illustrates the bi-contextual attention (BCA) module to estimate 3D translation. Specifically, relation-aware attention is designed to consider the relevance between objects based on a multi-head self-attention mechanism. Thus, the relation-aware feature $\mathbf{X}_r$ is given by

$$\mathbf{X}_r = \mathsf{MSA}(\mathsf{LN}(\mathbf{X}_o)) \tag{5}$$

On the other hand, scene-aware attention is presented to exploit scene context from global features. More precisely, we design it based on the MCA mechanism in which object features give queries and global features give keys and values:

$$\mathbf{X}_c = \mathsf{MCA}(\mathsf{LN}(\mathbf{X}_o), \mathsf{LN}(\mathbf{X}_g)) \tag{6}$$

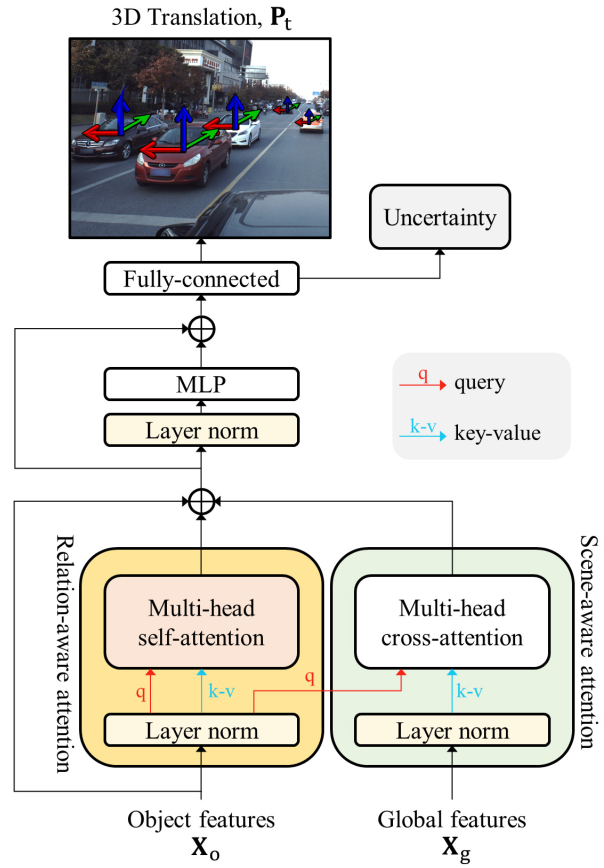
3D Translation, $\mathbf{P}_\mathrm{t}$
Uncertainty
Fully-connected
MLP
Layer norm
q query
k-v key-value
Relation-aware attention
Scene-aware attention
Multi-head self-attention
Multi-head cross-attention
q
k-v
q
k-v
Layer norm
Layer norm
Object features $\mathbf{X}_\mathrm{o}$
Global features $\mathbf{X}_\mathrm{g}$

Figure 4. Bi-contextual attention module for 3D translation estimation. Relation-aware attention extracts object's relative information from the object's individual features, while scene-aware attention interfuses various scene contexts into each object.

where $\mathbf{X}_\mathrm{c}$ is the scene-aware feature. Next, we construct translation features $\mathbf{X}_\mathrm{t}$ by integrating these external cues into object features:

$$\tilde{\mathbf{X}}_\mathrm{t} = \mathbf{X}_\mathrm{o} + \mathbf{X}_\mathrm{r}\mathbf{\Lambda}_\mathrm{r} + \mathbf{X}_\mathrm{c}\mathbf{\Lambda}_\mathrm{c} \quad (7)$$

$$\mathbf{X}_\mathrm{t} = \tilde{\mathbf{X}}_\mathrm{t} + \mathsf{MLP}(\mathsf{LN}(\tilde{\mathbf{X}}_\mathrm{t}))\mathbf{\Lambda}_\mathrm{t} \quad (8)$$

where $\mathbf{\Lambda}_\mathrm{r}$, $\mathbf{\Lambda}_\mathrm{c}$, $\mathbf{\Lambda}_\mathrm{t}$ are $c \times c$ learnable diagonal matrices [47] to scale the contribution of feature channels. Given translation features $\mathbf{X}_\mathrm{t}$, the last fully-connected layer regresses 3D translations $\mathbf{P}_\mathrm{t}$.

3.6. 3D Non-Maximum Suppression

Figure 5 shows a failure example of standard non-maximum suppression (NMS) in 2D image space. However, it also shows that spurious detection can be removed in Bird's eye view (BEV). Here, we propose a simple 3D NMS algorithm working on BEV as a post-processing step. As shown in Figure 5, our NMS identifies duplicated detection by comparing the x and z distances with thresholds λ_x and λ_z. It then iterates the procedure, which selects one detection with the highest detection score and removes duplicated detections.

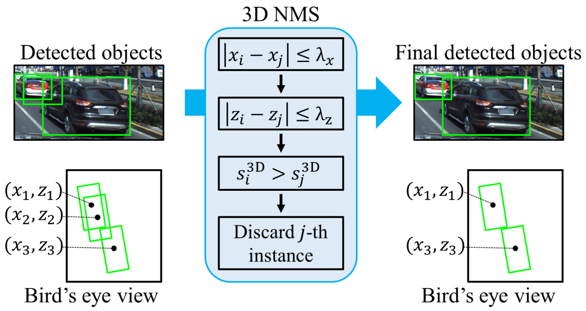

Figure 5. 3D non-maximum suppression to remove spurious objects based on their Bird-Eye-view distance.

For accurate 3D NMS, we introduce the 3D detection score based on depth uncertainty [33]. Note that restoring depth by inverting the image formation process is an ill-posed problem. And there exists inevitable uncertainty in estimated depth. Thus, it is essential to consider this uncertainty to make a certain decision using estimated depth. Specifically, we predict depth uncertainty u through the last fully-connected layer in Figure 4. We then define the 3D detection score considering depth uncertainty as

$$s^{\mathrm{3D}} = s^{\mathrm{2D}} \cdot \exp(-u) \quad (9)$$

where s^{2D} is 2D detection score by Mask R-CNN [16].

3.7. Loss functions

Regression losses. We define the translation loss L_{trans} as

$$L_{\mathrm{trans}} = |x - \hat{x}| + |y - \hat{y}| + \frac{\sqrt{2}}{u}|z - \hat{z}| + \log(u) \quad (10)$$

where $\hat{x}$, $\hat{y}$, $\hat{z}$ are ground-truth translation coordinate. The last two terms are the uncertainty regression loss [7, 33]. Note that due to inevitable uncertainty in depth estimation, difficult or noise-labeled objects often produce large errors causing unstable training. The uncertainty regression loss prevents it by inducing larger uncertainty for such cases. Also, we employ the rotation loss L_{rot} in [21], which constrains the range of rotation $\mathbf{p}_r$ to $[-\pi, \pi]$.

$$L_{\mathrm{rot}} = \begin{cases} |\mathbf{p}_\mathrm{r} - \hat{\mathbf{p}}_\mathrm{r}| & \text{if } |\mathbf{p}_\mathrm{r} - \hat{\mathbf{p}}_\mathrm{r}| \le \pi \\ |2\pi - |\mathbf{p}_\mathrm{r} - \hat{\mathbf{p}}_\mathrm{r}|| & \text{if } |\mathbf{p}_\mathrm{r} - \hat{\mathbf{p}}_\mathrm{r}| > \pi \end{cases} \quad (11)$$

where $\hat{\mathbf{p}}_\mathrm{r}$ is ground-truth rotation vector. For the shape L_{shape} loss, we simply adopt L2-loss between predicted and ground-truth ones.

Detection loss. The detection loss L_{det} constrains inaccurate bounding boxes and keypoints. More specifically, we define it as

$$L_{\mathrm{det}} = L_{\mathrm{rpn}} + L_{\mathrm{bbox}} + L_{\mathrm{kpts}} \quad (12)$$

Method	Detailed shape	A3DP-Abs			A3DP-Rel		
		mean	c-l	c-s	mean	c-l	c-s
DeepMANTA [5]	✗	20.10	30.69	23.76	16.04	23.76	19.80
Keypoints-based [43]	✗	20.40	31.68	**24.75**	16.53	24.75	19.80
3D-RCNN [24]	✓	16.44	29.70	19.80	10.79	17.82	11.88
Direct-based [43]	✓	15.15	28.71	17.82	11.49	17.82	11.88
GSNet [21]	✓	18.91	37.42	18.36	20.21	40.50	19.85
BAAM-ResNet101	✓	23.80	45.62	21.92	21.00	44.24	17.88
BAAM-Res2Net	✓	**25.19**	**47.31**	23.13	**22.85**	**46.21**	**20.31**

Table 1. Performance comparison with state-of-the-art methods for the monocular 3D pose and shape reconstruction on ApolloCar3D dataset [43]. No detailed shape methods use retrieval strategy, which searches a 3D shape to best match its 2D observation. We highlight the best and second best results in **bold** and underline.

where L_{rpn}, L_{bbox}, L_{kpts} are standard losses in [16] for RPN, 2D box head, and 2D keypoints head, respectively.

3D space loss. Even though objects' translation, rotation, and shape are interdependent in 3D space, the regression losses consider them independently. To emphasize their structure, we employ the 3D loss [13] that penalizes the mean vertex error between predicted and ground-truth vertices on the world space. Let $\mathbf{R} \in \mathbb{R}^{3\times3}$ be the rotation matrix corresponding to the estimated object rotation $\mathbf{p}_{\text{r}}$. Using the rotation matrix and the translation vector $\mathbf{t} = \mathbf{p}_{\text{t}}$, we can transform 3D mesh vertices in the camera space to the world space by $\mathbf{m}_{\text{world}} = \mathbf{R}\mathbf{m}^* + \mathbf{t}$. Here, $\mathbf{m}^* \in \mathbb{R}^{3\times v}$ is the reshaped matrix of the estimated 3D mesh $\mathbf{m}$. Moreover, we consider additional 3D spaces, rotation, and translation spaces, to clarify the ambiguous contributions of the translation and rotation. Thus, we define 3D mesh vertices in these spaces by $\mathbf{m}_{\text{rot}} = \mathbf{R}\mathbf{m}^*$ and $\mathbf{m}_{\text{trans}} = \mathbf{m}^* + \mathbf{t}$. Then, the 3D loss is given by

$$L_{\text{3D}} = \sum_{l \in \mathcal{S}} |\mathbf{m}_l - \hat{\mathbf{m}}_l| \tag{13}$$

where $\mathcal{S} = \{\text{world}, \text{trans}, \text{rot}\}$ is the 3D space set.

4. Experiments

4.1. Datasets and Metrics

ApolloCar3D [43]. It contains 4036, 200, and 1041 high-resolution images for training, validation, and testing. However, we only use the training and validation sets for our experiments because the ApolloCar3D test server is not serviced. ApolloCar3D images contain an average of 11.7 car objects described by 2D keypoints, 3D translation, and rotation labels. Each object is one of 79 car classes (*e.g*. sedan, coupe, SUV, and so on). For the ground-truth car shape, we adopt 3D mesh models in [21]. For the detection loss, we define a pseudo 2D bounding box since ApolloCar3D does not provide a 2D bounding box label. Specifically, we project object meshes to image space using camera parameters and their 3D translation and rotation. We then define a tight 2D box surrounding a projected mesh mask, as the pseudo 2D bounding box.

KITTI [12]. It is the widely used dataset in monocular 3D object detection. It consists of 7,481 images for training and 7,518 images for testing. As done in [6], we split the training data into a training set (3,712 images) and a validation set (3,769 images). Then, we conduct experiments on this split to validate the scalability of the bi-contextual attention module on 3D object detection.

Evaluation Metrics. For ApolloCar3D experiments, we adopt the average 3D precision (A3DP) [43], which jointly measures 3D translation, rotation, and shape reconstruction accuracy. According to the 3D translation error measurement scheme, we denote A3DP metrics with an absolute translation error and a relative one as A3DP-Abs and A3DP-Rel, respectively.

4.2. Implementation Details

Inference. We employ Res2Net [11] as our backbone, pre-trained on the COCO 2017 dataset [28]. We then perform 2D box and keypoints detection through Mask R-CNN [16]. After that, we estimate the 3D pose and shape for all detections. Finally, we conduct the proposed 3D NMS to remove spurious detections.

Training. We train the proposed BAAM network in two stages. First, we train BAAM with the detection loss L_{det} for 10 epochs. For this stage, we employ AdamW optimizer [32] with a learning rate of 0.0001. Second, we train BAAM with the total loss $L = L_{\text{det}} + L_{\text{trans}} + L_{\text{rot}} + L_{\text{shape}} + L_{\text{3D}}$ for 30 epochs. Here, we balance the contribution of losses with scales of 1, 0.5, 1, 3, and 0.01, respectively. Also, we use AdamW optimizer with an initial learning rate of 0.0001 and divide it by 10 at the 20 epoch. For the training, we use a mini-batch of size 4. The training is performed with an RTX A6000 GPU.

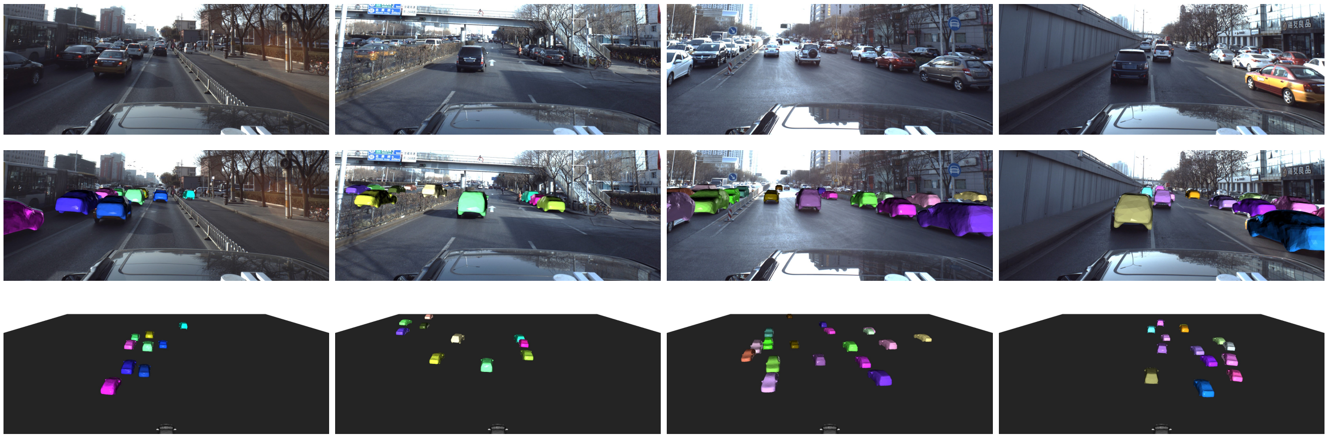

Figure 6. Qualitative results of BAAM on ApolloCar3D. Note how precisely BAAM estimates car 3D pose and shape.

Method	3D@IOU=0.7			BEV@IOU=0.7			3D@IOU=0.5			BEV@IOU=0.5		
	Easy	Mod.	Hard	Easy	Mod.	Hard	Easy	Mod.	Hard	Easy	Mod.	Hard
GUPNet [33]	23.18	16.24	13.57	30.18	22.42	**19.31**	58.99	43.64	39.34	65.16	48.88	42.93
+ BCA	24.23	16.48	13.60	30.90	22.58	19.21	60.32	**45.67**	**39.51**	**66.23**	49.73	43.23
+ BCA + 3D NMS	**24.33**	**16.55**	**13.66**	**31.00**	**22.66**	19.26	**60.60**	44.66	38.29	65.36	**49.99**	**43.40**
DID-M3D [40]	25.41	17.07	14.05	33.94	23.22	19.52	64.47	48.32	41.75	70.34	52.34	45.47
+ BCA	25.96	17.66	14.57	33.97	24.21	**20.69**	64.56	48.82	42.49	70.73	53.00	**47.75**
+ BCA + 3D NMS	**26.02**	**17.72**	**14.62**	**34.04**	**24.31**	19.87	**64.69**	**48.94**	**42.52**	**70.85**	**53.12**	46.36
DEVIANT [23]	24.58	16.52	14.50	32.63	23.05	20.00	60.97	44.78	40.18	65.31	49.63	43.50
+ BCA	25.34	16.98	14.93	32.77	23.21	20.12	61.11	**46.13**	40.31	65.45	49.80	**43.74**
+ BCA + 3D NMS	**25.40**	**17.03**	**14.95**	**32.80**	**23.32**	**20.15**	**61.29**	45.01	**40.34**	**65.64**	**49.99**	43.73

Table 2. The effectiveness of bi-contextual attention (BCA) and 3D non-maximum suppression (NMS). We add BCA module before the depth bias module of GUPNet [33] and DEVIANT [23], and attribute depth module of DID-M3D [40]. Also, we adopt 3D NMS in the post-processing steps of all. For the baselines, we reproduce results with the officially released code and parameters. We highlight the best and second best results in **bold** and underline.

4.3. Main Results

Results on ApolloCar3D. Table 1 compares the proposed BAAM with recent state-of-the-art algorithms on Apollo3D: DeepMANTA [5], Keypoints-based [43], 3D-RCNN [24], Direct-based [43], and GSNet [21]. Note that the proposed BAAM significantly outperforms the existing methods across the evaluation metrics. Compared to GSNet, BAAM improves A3DP-Abs and A3DP-Rel scores by 33% and 13%, respectively. For fair comparisons, we also report the performance of BAAM with ResNet101 [17] backbone, which is the same one with GSNet [21]. As shown in Table 1, BAAM-ResNet101 still exceeds GSNet with the improvement of 26% and 4% in terms of A3DP-Abs mean and A3DP-Rel mean. Figure 6 shows qualitative results on ApolloCar3D. The first row is the input images, and the second row is the result of our BAAM. The third row shows the reconstructed 3D space of BAAM in a different viewpoint. Corresponding car instances are depicted in the same color. We see that BAAM faithfully places car objects onto 3D space and estimates detailed object shapes.

Results on KITTI. We verify the scalability of the proposed BCA module and 3D NMS algorithm. To this end, we integrate our module into state-of-the-art algorithms for monocular 3D object detection: GUPNet [33], DID-M3D [40], and DEVIANT [23]. Table 2 reports the performance of these algorithms on KITTI *validation* set. Note that our BCA module improves their performances in all settings with only one exception. In addition, our 3D NMS algorithm further increases the overall performance of monocular 3D object detectors.

4.4. Ablation Study

Next, we study the contribution of three components: Attention-guided modeling (AGM); Bi-contextual attention (BCA); 3D non-maximum suppression (3D NMS). For the

Method	A3DP-Abs			A3DP-Rel			Rotate	Trans	Mesh
	Mean	c-l	c-s	Mean	c-l	c-s	Error	Error	IOU
Baseline	21.08	41.98	18.36	16.68	36.29	12.84	14.00	7.35	84.71
Baseline + AGM	22.80	43.01	21.94	18.79	38.97	15.54	13.96	7.23	85.55
Baseline + AGM + BCA	24.74	45.86	23.21	22.42	45.15	20.04	11.96	6.33	85.41
Baseline + AGM + BCA + 3D NMS	25.19	47.31	23.13	22.85	46.21	20.31	11.96	6.33	85.41

Table 3. Results on ApolloCar3D with different combinations of BAAM components: AGM, BCA, and 3D NMS. Baseline directly regresses mesh vertices and translation. We evaluate rotation error, translation error, and rendered mesh IOU with the ground truth boxes.

Method	Mesh Error	Mesh IOU
Regression	8.58	82.61
PCA-basis	7.25	85.08
Divide-and-conquer [21]	7.71	84.60
AGM	6.82	85.41

Table 4. Results on ApolloCar3D with different shape estimation methods. We measure rendered IOU, L1 distance error with respect to the ground truth mesh. The L1 distance error is in units of 10^{-2}.

ablation study, we design the baseline that excludes these components from BAAM and directly regresses 3D pose and shape through a single fully-connected layer. Table 3 reports the main results of ablation study on ApolloCar3D.

Impacts of AGM. In Table 3, AGM generates more accurate 3D mesh by improving mesh IOU +0.76 from Baseline. With a precise shape estimation, AGM increases A3DP-Abs mean, c-l, and c-s by +1.71, +3.25, and +2.8, respectively. Note that this result supports the importance of detailed shape estimation. For a comprehensive analysis, we replace AGM from our BAAM with different shape estimation methods: Regression, PCA-basis, and divide-and-conquer method [21]. The implementation details about these alternative methods are available in the supplementary material. Table 4 shows that the proposed AGM considerably exceeds the other methods with the highest mesh IOU and the lowest mesh error.

Impacts of BCA. As shown in Table 3, BCA significantly boosts A3DP-Abs and A3DP-Rel scores. This is because BCA effectively reduces translation errors by exploiting external object structures. Table 5 shows the contributions of relation-aware attention and scene-aware attention. We observe that both A3DP-Abs and A3DP-Rel scores are significantly degraded if one of them is missing. This demonstrates that the perception of both inter-object and road environments is critical to the 3D translation reasoning.

Impacts of 3D NMS. Table 3 shows that our 3D NMS further improves A3DP scores on ApolloCar3D. Also, we see that most best scores of state-of-the-arts on KITTI are obtained with our 3D NMS. Both experiments demonstrate the effectiveness of our 3D NMS. However, the improvement on ApolloCar3D is relatively larger than KITTI. This is because the ApolloCar3D has far more cars (11.7) than the KITTI (4.8) per image.

Method	A3DP-Abs			A3DP-Rel		
	Mean	c-l	c-s	Mean	c-l	c-s
RA	23.95	46.29	21.59	20.59	43.20	17.15
SA	23.66	45.59	21.84	20.59	42.64	17.15
RA+SA	25.19	47.31	23.13	22.85	46.21	20.31

Table 5. Results on ApolloCar3D with different combinations of BCA components. RA and SA mean the relation-aware attention and the scene-aware attention.

5. Conclusion

We proposed a novel algorithm, called BAAM, for monocular 3D pose and shape reconstruction. The main contributions of BAAM are bi-contextual attention, attention-guided modeling, and 3D NMS algorithm. Given various 2D primitives, BAAM reconstructs the object's shape as a mesh based on attention-guided modeling, which exploits relevance between individual objects and vehicle shape priors. Then, BAAM estimates the object's pose using the carefully designed bi-contextual attention module to consider relation-context inter objects and scene-context between the object and road environment. Finally, the 3D NMS algorithm eliminates spurious objects based on Bird-Eye-View geometry. Experiments demonstrated that BAAM significantly outperforms conventional algorithms on ApolloCar3D, and that BAAM improves monocular 3D object detectors on KITTI as a plugged-in-plug module.

Ackowledgements. This work was supported partly by the National Research Foundation of Korea (NRF) grants (NRF-2021R1A4A1031864, NRF-2022R1I1A3069113) and partly by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.RS-2022-00155857, Artificial Intelligence Convergence Innovation Human Resources Development (Chungnam National University)).

References

[1] Thiemo Alldieck, Gerard Pons-Moll, Christian Theobalt, and Marcus Magnor. Tex2shape: Detailed full human body geometry from a single image. In *ICCV*, pages 2293–2303, 2019. 2, 3

[2] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016. 4

[3] Xuyang Bai, Zeyu Hu, Xinge Zhu, Qingqiu Huang, Yilun Chen, Hongbo Fu, and Chiew-Lan Tai. Transfusion: Robust lidar-camera fusion for 3d object detection with transformers. In *CVPR*, pages 1090–1099, 2022. 1

[4] Garrick Brazil and Xiaoming Liu. M3d-rpn: Monocular 3d region proposal network for object detection. In *ICCV*, pages 9287–9296, 2019. 1

[5] 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*, pages 2040–2049, 2017. 2, 6, 7

[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. *NeurIPS*, 28, 2015. 6

[7] Yongjian Chen, Lei Tai, Kai Sun, and Mingyang Li. Monopair: Monocular 3d object detection using pairwise spatial relationships. In *CVPR*, pages 12093–12102, 2020. 1, 2, 5

[8] Hongsuk Choi, Gyeongsik Moon, JoonKyu Park, and Kyoung Mu Lee. Learning to estimate robust 3d human mesh from in-the-wild crowded scenes. In *CVPR*, pages 1475–1484, 2022. 2

[9] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2020. 4

[10] Emeç Erçelik, Ekim Yurtsever, Mingyu Liu, Zhijie Yang, Hanzhen Zhang, Pınar Topçam, Maximilian Listl, Yılmaz Kaan Çaylı, and Alois Knoll. 3d object detection with a self-supervised lidar scene flow backbone. In *ECCV*, 2022. 1

[11] Shang-Hua Gao, Ming-Ming Cheng, Kai Zhao, Xin-Yu Zhang, Ming-Hsuan Yang, and Philip Torr. Res2net: A new multi-scale backbone architecture. *IEEE TPAMI*, 43(2):652–662, 2019. 6

[12] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *CVPR*, pages 3354–3361, 2012. 2, 6

[13] Qichuan Geng, Hong Zhang, Feixiang Lu, Xinyu Huang, Sen Wang, Zhong Zhou, and Ruigang Yang. Part-level car parsing and reconstruction in single street view images. *IEEE TPAMI*, 44(8):4291–4305, 2021. 6

[14] 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*, pages 1080–1089, 2022. 2

[15] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003. 3

[16] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, pages 2961–2969, 2017. 3, 5, 6

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 7

[18] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016. 4

[19] Kuan-Chih Huang, Tsung-Han Wu, Hung-Ting Su, and Winston H Hsu. Monodtr: Monocular 3d object detection with depth-aware transformer. In *CVPR*, pages 4012–4021, 2022. 1

[20] Angjoo Kanazawa, Michael J Black, David W Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *CVPR*, pages 7122–7131, 2018. 2

[21] Lei Ke, Shichao Li, Yanan Sun, Yu-Wing Tai, and Chi-Keung Tang. Gsnet: Joint vehicle pose and shape reconstruction with geometrical and scene-aware supervision. In *ECCV*, pages 515–532, 2020. 2, 3, 4, 5, 6, 7, 8

[22] Muhammed Kocabas, Chun-Hao P Huang, Joachim Tesch, Lea Müller, Otmar Hilliges, and Michael J Black. Spec: Seeing people in the wild with an estimated camera. In *ICCV*, pages 11035–11045, 2021. 2

[23] Abhinav Kumar, Garrick Brazil, Enrique Corona, Armin Parchami, and Xiaoming Liu. Deviant: Depth equivariant network for monocular 3d object detection. In *ECCV*, 2022. 7

[24] Abhijit Kundu, Yin Li, and James M Rehg. 3d-rcnn: Instance-level 3d object reconstruction via render-and-compare. In *CVPR*, pages 3559–3568, 2018. 2, 6, 7

[25] Buyu Li, Wanli Ouyang, Lu Sheng, Xingyu Zeng, and Xiaogang Wang. Gs3d: An efficient 3d object detection framework for autonomous driving. In *CVPR*, pages 1019–1028, 2019. 1

[26] Peiliang Li, Xiaozhi Chen, and Shaojie Shen. Stereo r-cnn based 3d object detection for autonomous driving. In *CVPR*, pages 7644–7652, 2019. 1

[27] Peixuan Li, Huaici Zhao, Pengfei Liu, and Feidao Cao. Rtm3d: Real-time monocular 3d detection from object keypoints for autonomous driving. In *ECCV*, pages 644–660, 2020. 1

[28] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, pages 740–755, 2014. 6

[29] Zechen Liu, Zizhang Wu, and Roland Tóth. Smoke: Single-stage monocular 3d object detection via keypoint estimation. In *CVPRW*, pages 996–997, 2020. 1

[30] Zongdai Liu, Dingfu Zhou, Feixiang Lu, Jin Fang, and Liangjun Zhang. Autoshape: Real-time shape-aware monocular 3d object detection. In *ICCV*, pages 15641–15650, 2021. 1

[31] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. *ACM TOG*, 34(6):1–16, 2015. 2

[32] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 6

[33] 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*, pages 3111–3121, 2021. 1, 2, 5, 7

[34] Qianli Ma, Jinlong Yang, Anurag Ranjan, Sergi Pujades, Gerard Pons-Moll, Siyu Tang, and Michael J Black. Learning to dress 3d people in generative clothing. In *CVPR*, pages 6469–6478, 2020. 2, 3

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

[36] 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 *CVPR*, 2019. 2

[37] Fabian Manhardt, Wadim Kehl, and Adrien Gaidon. Roi-10d: Monocular lifting of 2d detection to 6d pose and metric shape. In *CVPR*, pages 2069–2078, 2019. 2

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

[39] Dennis Park, Rares Ambrus, Vitor Guizilini, Jie Li, and Adrien Gaidon. Is pseudo-lidar needed for monocular 3d object detection? In *ICCV*, pages 3142–3152, 2021. 2

[40] Liang Peng, Xiaopei Wu, Zheng Yang, Haifeng Liu, and Deng Cai. Did-m3d: Decoupling instance depth for monocular 3d object detection. In *ECCV*, 2022. 1, 2, 7

[41] Zequn Qin and Xi Li. Monoground: Detecting monocular 3d objects from the ground. In *CVPR*, pages 3793–3802, 2022. 1, 2

[42] Shaoshuai Shi, Chaoxu Guo, Li Jiang, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li. Pv-rcnn: Point-voxel feature set abstraction for 3d object detection. In *CVPR*, pages 10529–10538, 2020. 1

[43] Xibin Song, Peng Wang, Dingfu Zhou, Rui Zhu, Chenye Guan, Yuchao Dai, Hao Su, Hongdong Li, and Ruigang Yang. Apollocar3d: A large 3d car instance understanding benchmark for autonomous driving. In *CVPR*, pages 5452–5462, 2019. 2, 3, 4, 6, 7

[44] Jiaming Sun, Linghao Chen, Yiming Xie, Siyu Zhang, Qinhong Jiang, Xiaowei Zhou, and Hujun Bao. Disp r-cnn: Stereo 3d object detection via shape prior guided instance disparity estimation. In *CVPR*, pages 10548–10557, 2020. 1

[45] Yu Sun, Qian Bao, Wu Liu, Yili Fu, Michael J Black, and Tao Mei. Monocular, one-stage, regression of multiple 3d people. In *ICCV*, pages 11179–11188, 2021. 2

[46] Yu Sun, Wu Liu, Qian Bao, Yili Fu, Tao Mei, and Michael J Black. Putting people in their place: Monocular regression of 3d people in depth. In *CVPR*, pages 13243–13252, 2022. 2

[47] Hugo Touvron, Matthieu Cord, Alexandre Sablayrolles, Gabriel Synnaeve, and Hervé Jégou. Going deeper with image transformers. In *ICCV*, 2021. 5

[48] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 4

[49] 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. 1, 2

[50] Yunpeng Zhang, Jiwen Lu, and Jie Zhou. Objects are different: Flexible monocular 3d object detection. In *CVPR*, pages 3289–3298, 2021. 1, 2

[51] Yunpeng Zhang, Wenzhao Zheng, Zheng Zhu, Guan Huang, Dalong Du, Jie Zhou, and Jiwen Lu. Dimension embeddings for monocular 3d object detection. In *CVPR*, pages 1589–1598, 2022. 1, 2