*Supplementary Material for*

# 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

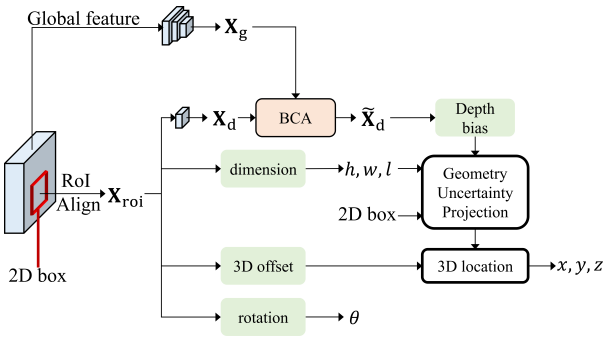


Figure S-1. 3D estimation of GUPNet with the proposed BCA module.

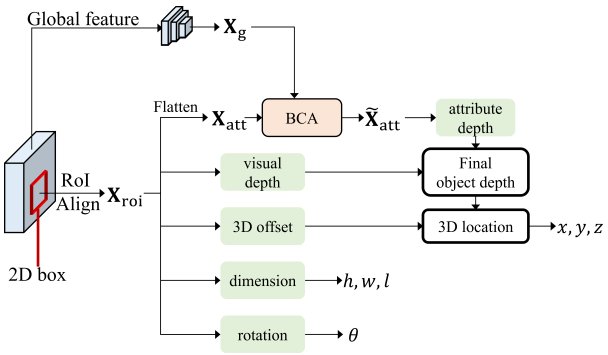


Figure S-2. 3D estimation of DID-M3D with the proposed BCA module.

## S-1. Network architecture for experiments on the KITTI dataset

To validate the effectiveness of bi-contextual attention (BCA) for the monocular 3D object detection on the KITTI dataset, we add the BCA module into three mature methods: GUPNet [4], DEVIANT [3], and DID-M3D [5]. These three methods extract RoI features $\mathbf{X}_{\text{roi}} \in \mathbb{R}^{n\times r\times 7\times 7}$ from the backbone feature map using RoIAlign [1], where $r$ is a channel dimension of $n$ objects. $\mathbf{X}_{\text{roi}}$ passes through multiple parallel heads for 3D estimation. We describe how we integrate the proposed BCA with each method in the paragraphs below.

**GUPNet with BCA.** Figure S-1 illustrates 3D estimation of GUPNet with the proposed BCA module. We adopt the BCA module before the depth bias module in GUPNet. Specifically, we process RoI features $\mathbf{X}_{\text{roi}}$ using one convolution operation and global average pooling to obtain a feature matrix $\mathbf{X}_{\text{d}} \in \mathbb{R}^{n\times d}$, where $d = 256$ is a feature dimension. Also, we extract global features $\mathbf{X}_{\text{g}} \in \mathbb{R}^{g\times d}$ by feeding the backbone feature map into three convolutional layers, global average pooling, and reshape operator as done in the proposed BAAM. The BCA module combines $\mathbf{X}_{\text{d}}$ and $\mathbf{X}_{\text{g}}$ to obtain bi-contextual depth features $\tilde{\mathbf{X}}_{\text{d}} \in \mathbb{R}^{n\times d}$. Unlike the depth bias module takes $\mathbf{X}_{\text{d}}$ as the input in the original GUPNet, the depth bias module in GUPNet with BCA takes $\tilde{\mathbf{X}}_{\text{d}}$ to yield the depth bias and uncertainty. Then, the depth bias and uncertainty are used for 3D estimation as done in GUPNet.

**DEVIANT with BCA.** DEVIANT has a similar process to GUPNet for 3D estimation. DEVIANT also contains the depth bias module and feeds $\mathbf{X}_{\text{d}}$ into the depth bias module. In this work, as done in GUPNet with BCA, we add BCA before the depth bias module to obtain $\tilde{\mathbf{X}}_{\text{d}}$. Then, $\tilde{\mathbf{X}}_{\text{d}}$ is fed into the depth bias module for 3D estimation.

**DID-M3D with BCA.** As shown in Figure S-2, we add the BCA module before the attribute depth module in DID-M3D. Specifically, we first flatten the RoI features $\mathbf{X}_{\text{roi}}$ to construct attribute features $\mathbf{X}_{\text{att}} \in \mathbb{R}^{n\times \tilde{r}}$, where $\tilde{r} = r\times 7\times 7$ is a flattened feature dimension. We further extract global features $\mathbf{X}_{\text{g}} \in \mathbb{R}^{g\times \tilde{r}}$ from backbone feature maps

| Method | A3DP-Abs | | | A3DP-Rel | | | Rotate |
|---|---|---|---|---|---|---|---|
| | Mean | c-l | c-s | Mean | c-l | c-s | error |
| OF | 25.19 | 47.31 | 23.13 | 22.85 | 46.21 | 20.31 | 11.96 |
| BF | 23.66 | 45.59 | 21.84 | 20.59 | 42.64 | 17.15 | 12.64 |

Table S-1. Ablation study for 3D rotation estimation. 'OF' uses the object features to regress rotation, while 'BF' uses the output features of the BCA module.

through three convolutional layers, global average pooling, and reshape operator. Given the attribute features $\mathbf{X}_{\text{att}}$ and the global features $\mathbf{X}_{\text{g}}$, the BCA module aggregates them into the bi-contextual attribute features and restores their shape into $\tilde{\mathbf{X}}_{\text{att}} \in \mathbb{R}^{n\times r\times 7\times 7}$. The attribute depth module takes $\tilde{\mathbf{X}}_{\text{att}}$ to estimate depth maps for 3D estimation.

## S-2. Implementation details for the ablation study of AGM

For the ablation study of the proposed attention-guided modeling (AGM), we replace AGM with three different shape estimation methods: Regression, PCA-basis, and Divide-and-Conquer in GSNet [2]. We present detailed descriptions of these methods in the paragraphs below. For training, we follow the same strategy as the proposed BAAM.

**Regression.** Regression method directly estimates mesh vertices $\mathbf{m}$ from object features $\mathbf{X}_{\text{o}}$ using three fully-connected layers.

**PCA-basis.** PCA-basis method estimates the shape parameters $\beta \in \mathbb{R}^{10}$ from the object features $\mathbf{X}_{\text{o}}$ through three fully-connected layers. Then, the shape parameters are decoded to mesh vertices via $\mathbf{m} = \bar{\mathbf{m}}_{\text{s}} + \mathbf{P}\beta$, where $\mathbf{P} \in \mathbb{R}^{3v\times 10}$ is the PCA-basis. To construct PCA-basis $\mathbf{P}$, we implement PCA on the template offsets $\mathbf{O}_{\text{s}}^{T}$ and find the ten dimensional shape basis.

**Divide-and-Conquer.** Divide-and-Conquer method [2] transfers the object features $\mathbf{X}_{\text{o}}$ into three fully-connected layers to generate four PCA parameters $\boldsymbol{\beta}_{\text{div}} = [\beta^1_{\text{div}}, \beta^2_{\text{div}}, \beta^3_{\text{div}}, \beta^4_{\text{div}}]$, where $\beta^i_{\text{div}} \in \mathbb{R}^{b_i}$ is $i$-th shape parameters consisting of $b_i$ dimension. Then, they are decoded to four different meshes with PCA-bases. Finally, four decoded meshes are blended into estimated mesh $\mathbf{m}$ with their respective classification probabilities.

## S-3. Further experimental results

**3D rotation estimation based on BCA features.** The proposed BAAM simply regresses the object feature $\mathbf{X}_{\text{o}}$ to estimate 3D rotation $\mathbf{P}_{\text{r}}$. This is because the object features already encode the rich information for rotation, and the further process to interfuse external object structure may cause information loss. To validate this, we re-design BAAM to estimate 3D rotation using features extracted from the BCA module. As shown in Table S-1, output features of the BCA module degrade rotation performance in terms of both A3DP-Rel and A3DP-Abs. This indicates that the proposed BCA is ineffective for 3D rotation estimation, while it is essential for 3D translation estimation.

| Method | 3D@IOU=0.7 | | | BEV@IOU=0.7 | | |
|---|---|---|---|---|---|---|
| | Easy | Mod. | Hard | Easy | Mod. | hard |
| DEVIANT | **21.88** | 14.46 | 11.89 | 29.65 | 20.44 | 17.43 |
| DEVIANT + BCA | 21.82 | **14.52** | **11.99** | **30.03** | **20.75** | **17.60** |

Table S-2. Performance comparison of DEVIANT without and with the proposed BCA for the monocular 3D object detection on KITTI *test* set. We highlight the best results in **bold**.

| Method | A3DP-Abs | | | A3DP-Rel | | |
|---|---|---|---|---|---|---|
| | Mean | c-l | c-s | Mean | c-l | c-s |
| GSNet | 13.83 | 27.29 | 13.63 | 4.42 | 11.84 | 2.36 |
| BAAM | **20.73** | **36.63** | **22.19** | **8.17** | **18.98** | **5.99** |

Table S-3. Performance comparison of BAAM with GSNet on Pascal3D+ [6]. We highlight the best results in **bold**.

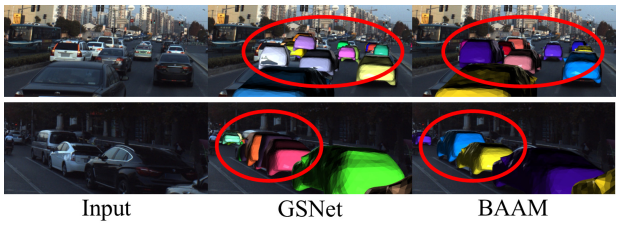


Figure S-3. Qualitative comparison of the proposed BAAM with GSNet. We use red ovals to emphasize the failure examples of GSNet.


Figure S-4. Visualization of relation-aware attention scores. Red texts are relation-aware attention scores for a car in a red box to the other cars.

**Experiments on KITTI *test* set.** We further report the performance of the proposed BCA module on KITTI *test* set in Table S-2. We observe that the BCA module improves the performance of DEVIANT [3] in all metrics except 'Easy' in '3D@IOU=0.7' metric.

**Results on another dataset.** Table S-3 compares BAAM with GSNet on the Pascal3D+ [6] dataset, which also provides both 3D pose and shape labels. We evaluate each performance with A3DP and observe that the proposed BAAM significantly outperforms GSNet on the Pascal3D+.

**More qualitative results**: Figure S-3 shows the qualitative comparison of the proposed BAAM with GSNet. BAAM estimates 3D pose and shape more precisely and eliminates duplicated detections more clearly than GSNet. Figure S-4 visualizes relation-aware attention scores for a car in a red box to the other cars. We see that front and distant objects, which are critical to the relative depth, present high attention scores for the relation-aware attention of BCA.

**Qualitative comparison of 3D NMS with 2D NMS.** Figure S-5 shows qualitative comparison of the proposed 3D NMS with the standard non maximum suppression (2D NMS). We can see that 3D NMS faithfully eliminates duplicated objects.

## References


[1] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, pages 2961–2969, 2017. 1

[2] 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] Abhinav Kumar, Garrick Brazil, Enrique Corona, Armin Parchami, and Xiaoming Liu. Deviant: Depth equivariant network for monocular 3d object detection. In *ECCV*, 2022. 1, 2

[4] 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

[5] 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

[6] Yu Xiang, Roozbeh Mottaghi, and Silvio Savarese. Beyond pascal: a benchmark for 3d object detection in the wild. In *WACV*, 2014. 2, 3

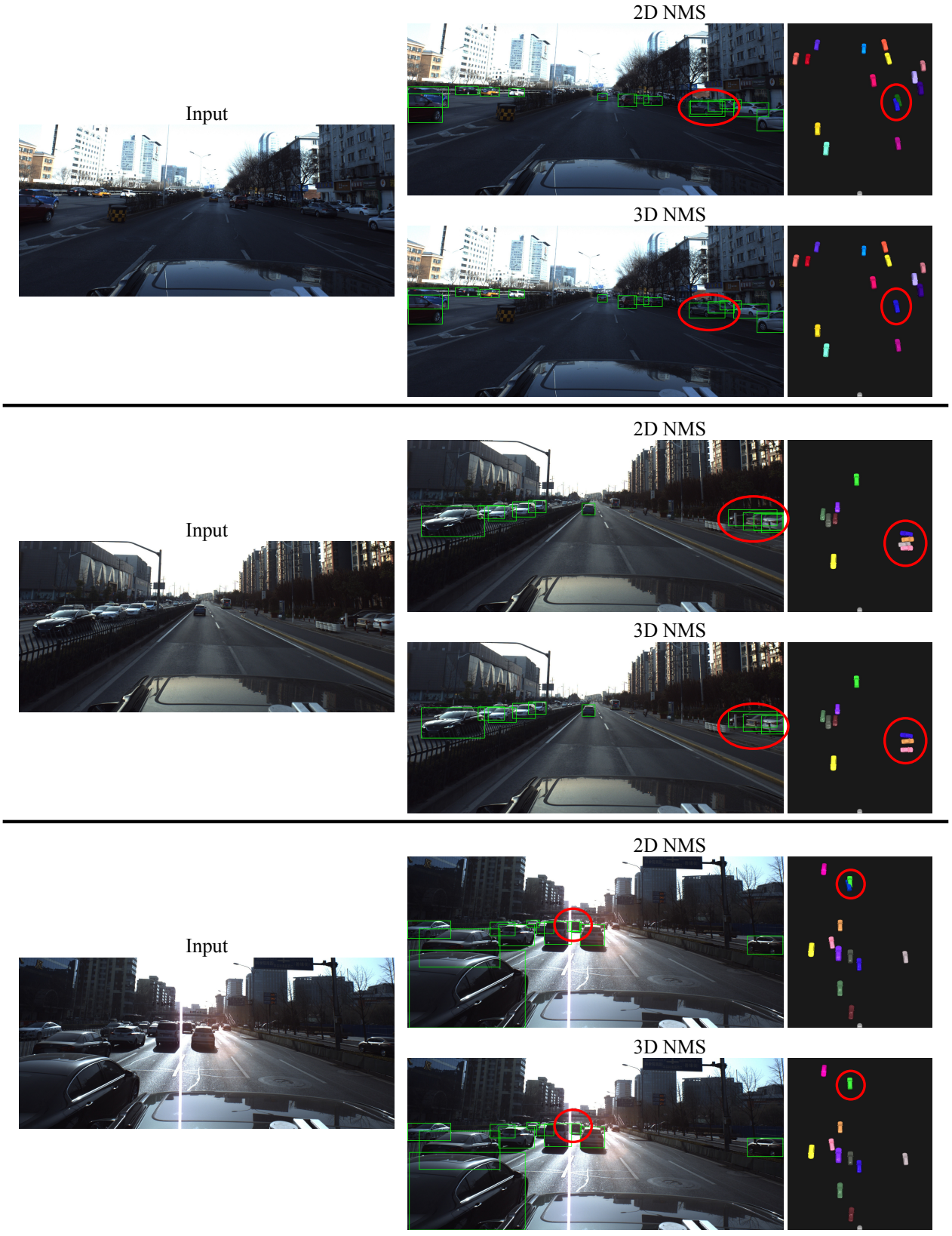


Figure S-5. Qualitative comparison of the proposed 3D NMS with the standard 2D NMS. The first column shows input images. The second column represents 2D bounding boxes after 2D NMS and 3D NMS, while the third column illustrates reconstructed 3D scenes in the Bird's-eye view. We use red ovals to emphasize the failure examples of 2D NMS.