Supplementary Material: Depth-conditioned Dynamic Message Propagation for Monocular 3D Object Detection

Li Wang[1*] Liang Du[1*] Xiaoqing Ye[2*] Yanwei Fu[1] Guodong Guo[2]
Xiangyang Xue[1] Jianfeng Feng[1] Li Zhang[1†]
[1]Fudan University [2]Baidu Inc.

1. Architecture details

As described in Section 3 in our paper, DDMP-3D contains image and depth feature encoding branches, with two DDMP modules adopted at Stage II and III, respectively. We demonstrate the architecture details in Table 1. Since two DDMP modules ("DDMP_1" and "DDMP_2") share a similar architecture, we only report the details of "DDMP_1". "DDMP_1" first integrates image features from stage II with depth features from stage II / III / IV (stage2_depth2 / 3 / 4) and then concatenate the outputs together. The outputs are scaled to the size of image features ("DDMP_1 (update)"). Note that the codes of constructing the model are attached in the supplementary material.

2. Additional experiments

Loss weight selection of auxiliary tasks. The loss weights for two branches in Equation 10 in our paper determine the influence of the auxiliary task on main task, which is a key hyper-parameter in our DDMP-3D framework. To explore the sensitivity of this parameter, we conduct experiments as shown in Table 2.

Paying more attention to the main task or at least equal weights to two tasks can achieve better performance for our model. When the weight for auxiliary task is equal to that of main task, it is favorable to detect objects on moderate and hard settings owing to its sensitivity to the centers. While it is friendly to detect easy objects with higher weight for main task. A relatively high weight to auxiliary task brings slightly negative effect on the final performance. This reflects that L_{det} is essential on detection results whose weight should not be less than that of L_{dep}.

Statistic analysis on 3D metric. To further demonstrate the effectiveness of the proposed CDE, we compare the errors on the specific metrics (center "xyz") of the baseline method with or without CDE.

As shown in Figure 1, we can see that our proposed CDE improves the baseline method in "x", "y" and "z", resulting in more accurate monocular 3D object detection. Note that the "x", "y", and "z" indicate the 3D camera coordinates of the object center point.

Ablation study on the auxiliary task for the depth encoding. We report the experiment results of deploying other auxiliary tasks in Table 3. It is observed that various auxiliary tasks have certain effects on the performances. The task of 3D center regression ("xyz") is critical, which introduces notable improvements on all settings.

However, the performance of adding 3D bounding box regression ("whl + rotation") or classification task experienced a drop at some settings. We consider that it is difficult for bounding box regression and classification on depth map without well-defined boundary and distinctive appearance. Therefore, we further validate the hypothesis that the center-aware depth feature encoding helps monocular 3D object detection.

Different message propagation strategies. How to effectively deliver the depth information through image feature domain and learn context- and depth-aware feature representation is critical for monocular 3D detection. This is also the objective of this paper. We perform different message propagation strategies to verify the effectiveness of our proposed DDMP-3D.

As shown in Table 4, "3DNet" is the baseline in D^{4}LCN [1], which only contains the single detection branch without the guidance of depth map. "3DNet w/ DGMN [2]" augments detection branch with the DGMN formulation to perform the effective feature learning in the RGB feature domain. "Baseline" integrates images with depth maps via a common multiplication operation. With the guidance of depth maps, it easily outperforms the above two methods. "3DNet w/ DGMN + Depth" introduces the depth informa-

*The first three authors contributed equally to this work.
†Li Zhang (lizhangfd@fudan.edu.cn) is the corresponding author with School of Data Science, Fudan University. Li Wang and Xiangyang Xue are with School of Computer Science, Fudan University. Yanwei Fu is with the School of Data Science, MOE Frontiers Center for Brain Science, and Shanghai Key Lab of Intelligent Information Processing, Fudan University. Liang Du and Jianfeng Feng are with the Institute of Science and Technology for Brain-Inspired Intelligence, Fudan University.

Table 1. Detailed architecture. The table expands the details of the DDMP process on image stage II (DDMP_1), including the message propagation from image stage II and depth stage II / III / IV, and the message updating on image stage II.

Module	Type / Stride	Input name	Output name: size
Detection backbone (ResNet-50)	conv1 / s=2	image	img_conv1: $64 \times 256 \times 880$
	conv2_x / s=2	img_conv1	img_stage1: $256 \times 128 \times 440$
	conv3_x / s=2	img_stage1	img_stage2: $512 \times 64 \times 220$
	conv4_x / s=2	img_stage2	img_stage3: $1024 \times 32 \times 110$
	conv5_x (dilated=2) / s=1	img_stage3	img_stage4: $2048 \times 32 \times 110$
Depth backbone (ResNet-50)	conv1 / s=2	estimated depth map	dep_conv1: $64 \times 256 \times 880$
	conv2_x / s=2	dep_conv1	dep_stage1: $256 \times 128 \times 440$
	conv3_x / s=2	dep_stage1	dep_stage2: $512 \times 64 \times 220$
	conv4_x / s=2	dep_stage2	dep_stage3: $1024 \times 32 \times 110$
	conv5_x (dilated=2) / s=1	dep_stage3	dep_stage4: $2048 \times 32 \times 110$
DDMP_1 (stage2_depth2)	conv 1×1	img_stage2	img_stage22: $256 \times 64 \times 220$
	conv 3×3	img_stage22	img_stage22_offset: $18 \times 64 \times 220$
	deform_unfold 3×3	img_stage22	img_stage22_sample: $256 \times 9 \times 64 \times 220$
	conv 1×1	dep_stage2	dep_stage22: $256 \times 64 \times 220$
	deform_conv $3 \times 3 (group = 1)$	dep_stage22	dep_stage22_affinity: $9 \times 64 \times 220$
	conv 3×3	dep_stage22	dep_stage22_filter: $9 \times 64 \times 220$
	dot	img_stage22_sample; dep_stage22_filter	stage22_sample: $256 \times 9 \times (64 * 220)$
	matmul(group=1)	stage22_sample; dep_stage22_affinity	message_stage22: $256 \times 64 \times 220$
DDMP_1 (stage2_depth3)	Similar to stage2_depth2 (+ interpolate on dep_stage3)	img_stage2; dep_stage3	message_stage23: $256 \times 64 \times 220$
DDMP_1 (stage2_depth4)	Similar to stage2_depth2 (+ interpolate on dep_stage4)	img_stage2; dep_stage4	message_stage24: $256 \times 64 \times 220$
DDMP_1 (update)	concat & conv 3×3	img_stage2; message_stage22/23/24	img_stage2: $512 \times 64 \times 220$
DDMP_2	Similar to DDMP_1	img_stage3; dep_stage2/3/4	img_stage3: $1024 \times 32 \times 110$

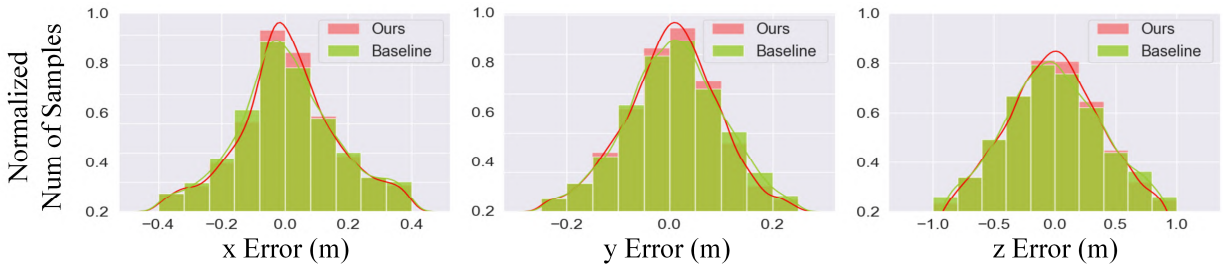

Figure 1. The statistic analysis and comparison of the baseline (green) and the baseline with our CDE (red). The vertical axis of the chart represents the number of samples after normalization. Improvements can be observed in the metrics "x", "y", and "z".

tion via the common multiplication operation. "DDMP" is our proposed module for integrating image and depth via graph message propagation.

The large gains on all settings demonstrate its effectiveness on propagating depth-conditioned messages. Different with DGMN [2], our proposed DDMP generates hybrid filters and affinities used for propagating message from the multi-scale sampled depth feature; "DDMP + CDE" augments the "CDE" task which has been discussed in the main paper. Note that thanks to the non-linear Softmax operation on the generated affinity matrix, the network learns from the normalized affinities to further boost the final detection performance, as is shown in the last two rows in Table 4.

Table 2. Comparison results (3D "Car" detection) of different weights of auxiliary tasks on val split set (IoU = 0.7). α and β are the weights for L_{det} and L_{dep}, respectively.

$\alpha : \beta$	AP_{3D}			AP_{BEV}		
	Mod.	Easy	Hard	Mod.	Easy	Hard
1:0	22.84	28.12	19.09	27.05	37.11	24.20
1:2	22.71	31.35	18.94	27.18	37.96	24.38
2:1	22.85	**32.32**	19.35	27.36	**41.65**	24.47
1:1	**23.13**	31.14	**19.45**	**27.46**	37.71	**24.53**

Table 3. Comparison results (3D "Car" detection) of different auxiliary tasks on val split set (IoU = 0.7). "DDMP + bbox", "DDMP + class", and "DDMP + center" stand for the bounding boxes regression, classification and center regression tasks, respectively.

Method	AP_{3D}			AP_{BEV}		
	Mod.	Easy	Hard	Mod.	Easy	Hard
DDMP	22.84	28.12	19.09	27.05	37.11	24.20
DDMP + bbox	22.48 (-0.36)	28.84 (+0.72)	18.31 (-0.78)	26.06 (-0.99)	36.35 (-0.76)	21.00 (-3.20)
DDMP + class	22.69 (-0.15)	28.72 (+0.60)	19.16 (+0.07)	26.94 (-0.09)	36.87 (-0.24)	24.11 (-0.09)
DDMP + center	**23.13** (+0.29)	**31.14** (+3.02)	**19.45** (+0.36)	**27.46** (+0.41)	**37.71** (+0.60)	**24.53** (+0.33)

Table 4. Comparison results (3D "Car" detection) of different message integration positions on val split set (IoU = 0.7).

Method	Image input	Depth map input	AP_{3D}			AP_{BEV}		
			Mod.	Easy	Hard	Mod.	Easy	Hard
3DNet	✓	-	14.61	17.94	12.74	19.89	24.87	16.14
3DNet w/ DGMN [2]			16.98	20.12	15.17	21.49	26.40	17.96
Baseline (3DNet + Depth)	✓	✓	18.82	26.03	16.27	24.18	33.06	19.63
3DNet w/ DGMN [2] +Depth			19.59	27.78	16.48	25.30	35.59	20.32
DDMP	✓	✓	22.84	28.12	19.09	27.05	37.11	24.20
DDMP + CDE			23.13	31.14	**19.45**	27.46	37.71	24.53
DDMP (*Softmax*) + CDE			**23.17**	**32.40**	19.35	**27.85**	**42.05**	**24.91**

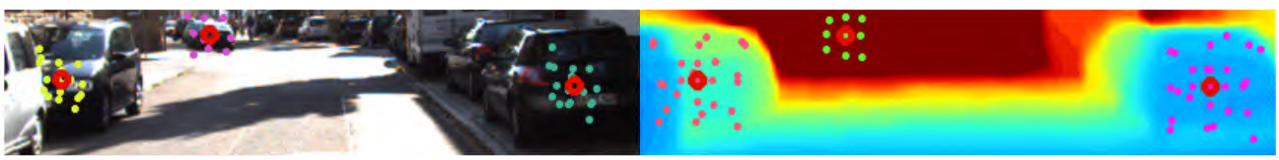

Figure 2. Visualization of sampling points on the images and depth maps, and the predicted results on the KITTI dataset.

Figure 3. More qualitative results on the KITTI dataset. The 3D ground-truth boxes and our DDMP-3D predictions are drawn in green and red, respectively.

3. Additional qualitative results

Visualization of dynamic sampling points. Figure 2 shows dynamic sampling points based on the learned Δd and $\Delta \tilde{d}$ from images and depth maps, respectively. The receiving nodes are shown with red circles. As shown in left figure, our sampled image nodes accurately perceive the semantic context: object boundary of left car and the small object, to enable more effective message passing. Also, in right figure, we demonstrate our dynamically sampled multi-scale depth nodes that dedicate to capture the context of the target objects.

More qualitative results. Figure 3 shows more qualitative results on the KITTI dataset. The 3D ground-truth boxes and our DDMP-3D predictions are drawn in green and red, respectively. As clearly observed, DDMP-3D can produce high-quality 3D bounding boxes in various scenes.

References

[1] 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*, 2020. 1

[2] Li Zhang, Dan Xu, Anurag Arnab, and Philip HS Torr. Dynamic graph message passing networks. In *CVPR*, 2020. 1, 2, 3